N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM MICROFICHE. ALTHOUGH IT IS RECOGNIZED THAT CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED IN THE INTEREST OF MAKING AVAILABLE AS MUCH INFORMATION AS POSSIBLE

"Made available under NASA sponsorship in the interest of early and wide dissemination of Earth Resources Survey Program information and without liability for any use made thereof."

778-10770
8.0-10889

JSC-13030
Volume 2
Part I NASA CR-160709

AS-BUILT DOCUMENT FOR THE CAMS IMAGE-100 HYBRID SYSTEM

Job Order 71-195

VOLUME 2

DETAILED FLOW CHARTS AND PROGRAM LISTINGS

(E80-10289) AS-BUILT DESIGN SPECIFICATION FOR THE CMAS IMAGE-100 HYBRID SYSTEM. VOLUME 2: DETAILED FLOW CHARTS AND PROGRAM LISTINGS, PART 1 (Lockheed Electronics Co.) 395 p HC A17/MF A01 CSCL 09B G3/45

N80-30857

Unclas 00289

Prepared By

Lockheed Electronics Company, Inc.
Systems and Services Division
Houston, Texas

Contract NAS 9-15200

For

EARTH OBSERVATIONS DIVISION

SCIENCE AND APPLICATIONS DIRECTORATE



National Aeronautics and Space Administration

LYNDON B. JOHNSON SPACE CENTER

Houston, Texas

August 1977

LEC-10822
Volume 2
Part I

JSC-13030
Volume 2
Part I

"AS-BUILT" DESIGN SPECIFICATION
FOR THE
CAMS IMAGE-100 HYBRID SYSTEM

Job Order 71-195

VOLUME 2
DETAILED FLOW CHARTS AND PROGRAM LISTINGS

Assembled By

L. E. Giddings

from contributions of the following persons:

R. T. Minter
K. L. Pattison
P. S. Lin
G. J. Champagne
E. J. Hightower
W. A. Holley
J. S. Huang
T. R. Kell
D. L. Loe
L. F. Robinson
R. M. Rodriquez
J. K. Rowland
C. D. Shih
H. G. Thadani
S. G. Thadani
B. R. Thompson
E. L. Wilson

APPROVED BY

P. L. Krumm, Supervisor
Applications Software Section

Prepared By

Lockheed Electronics Company, Inc.

For

Earth Observations Division

Science and Applications Directorate

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
LYNDON B. JOHNSON SPACE CENTER
HOUSTON, TEXAS

August 1977

LEC-10822
Volume 2
Part I

SPECIAL NOTE

The information in all three volumes of this document has been carefully checked. It is current at the time of publication, the end of August, 1977. This document will not be revised to show corrections and further changes. Rather, a new document will be issued toward the end of 1977 incorporating all changes, and making necessary corrections. The new volumes will be issued under the title: "As-Built Design Specifications for the CAMS Image-100 Hybrid System, as modified. The new document will be issued as LEC-11216 and JSC-13118.

Please bring errors and corrections to the attention of L. Giddings, 333-6311, mail code C42.

PRECEDING PAGE BLANK NOT FILMED

ABSTRACT

This document shows the CAMS Image-100 Hybrid System as it was actually built. Volume 1 lists the computer programs for each portion of the system, together with functional flow charts. Subroutines and function for each program are described in the summary. Volume 2 presents detailed flow charts and listings of all items listed in the first volume. The third volume presents brief descriptions and listings of subroutines shared by several programs. All three volumes close with an index of computer elements of the entire document.

PRECEDING PAGE BLANK-NOT FILMED

CONTENTS

Section		Page
1.	IMAGE DATA BASE UPDATE PROGRAM IMAUPD.	1-1
1.1	SUBROUTINE REPORT	1-8
1.2	SUBROUTINE DIRCRE	1-10
1.3	SUBROUTINE DELEAT	1-13
1.4	SUBROUTINE RDDISK	1-17
1.5	SUBROUTINE SRDISK	1-20
1.6	SUBROUTINE JULIAN	1-23
1.7	SUBROUTINE FILEST	1-25
1.8	SUBROUTINE RDHEAD	1-27
1.9	SUBROUTINE PRESET	1-29
1.10	SUBROUTINE STRAYS.	1-32
1.11	SUBROUTINE TAPSCN.	1-35
1.12	SUBROUTINE SQUEEZ.	1-37
1.13	SUBROUTINE IFLD.	1-38
1.14	SUBROUTINE KAUTH	1-40
1.15	SUBROUTINE T2DR.	1-42
2.	DO/DU FIELD UPDATE PROGRAM MDODU	2-1
2.1	MAIN PROGRAM FLDUP1	2-4
2.2	SUBROUTINE CNTRL.	2-11
2.3	SUBROUTINE EXPTD.	2-22
2.4	SUBROUTINE FLDST.	2-24
2.5	SUBROUTINE FIELD.	2-30

Section Page

2.6 SUBROUTINE DCOORD 2-33

2.7 SUBROUTINE FLDEND 2-37

2.8 SUBROUTINE SEGNED 2-40

2.9 SUBROUTINE RDDIR. 2-44

2.10 SUBROUTINE RDDODU. 2-46

2.11 SUBROUTINE RDDOT 2-48

2.12 SUBROUTINE WRDODU. 2-50

2.13 SUBROUTINE WRDOT 2-52

2.14 SUBROUTINE WRDIR 2-54

2.15 SUBROUTINE UPDOT 2-56

2.16 SUBROUTINE JULIAN. 2-59

2.17 SUBROUTINE FLDINT. 2-61

2.18 SUBROUTINE FLGDOT. 2-65

2.19 SUBROUTINE NFLDST. 2-68

2.20 SUBROUTINE NCNTRL. 2-70

2.21 SUBROUTINE NFIELD. 2-72

2.22 SUBROUTINE NSEGND. 2-75

3. DOT DATA UPDATE PROGRAM DOTUPD 3-1

3.1 FUNCTION UNPAK. 3-6

3.2 FUNCTION CATLOG 3-8

3.3 SUBROUTINE DIRLOD 3-10

3.4 FUNCTION JULIAN 3-12

3.5 DATA RANDOT 3-14

3.6 SUBROUTINE UPDATE 3-16

Section Page

3.7 SUBROUTINE ALLUPD 3-18

3.8 FUNCTION RDCARD 3-21

3.9 FUNCTION SKIP 3-23

3.10 FUNCTION CARDIN. 3-25

3.11 FUNCTION COMPAR. 3-27

3.12 SUBROUTINE SUBSTR. 3-29

3.13 SUBROUTINE OPMESS. 3-31

4. CAMS/CAS INTERFACE TAPE UPDATE PROGRAM BSTAT 4-1

4.1 SUBROUTINE CDRED. 4-9

4.2 SUBROUTINE IOPRNT 4-16

4.3 SUBROUTINE HEXD 4-18

5. CLASSIFICATION AND CLUSTER MAP UPDATE PROGRAM DTERM. . 5-1

5.1 SUBROUTINE DIRUPD 5-5

5.2 SUBROUTINE MAPUPD 5-7

5.3 SUBROUTINE CLASS 5-9

5.4 SUBROUTINE CLUSTR 5-11

5.5 SUBROUTINE FMAINT 5-13

5.6 SUBROUTINE SORTRC 5-15

5.7 SUBROUTINE OPMESS 5-17

5.8 SUBROUTINE JULIAN 5-19

5.9 SUBROUTINE HEADIN 5-21

5.10 SUBROUTINE REPORT. 5-24

5.11 DATA LOOKUP. 5-26

Section Page

5.12 SUBROUTINE ZNAME 5-28

6. SEGMENT DELETE PROGRAM SEGDEL. 6-1

6.1 SUBROUTINE ELAPSE 6-6

7. CONTROL PROGRAM CAMSEX 7-1

8. INITIATE SEGMENT ANALYSIS INIT 8-1

8.1 SUBROUTINE FTRNFR 8-16

8.2 SUBROUTINE INTLZE 8-18

9. IMAGE DISPLAY FULOI3 9-1

9.1 SUBROUTINE CRUNCH 9-10

9.2 SUBROUTINE GABI 9-11

9.3 SUBROUTINE COMLUT 9-12

9.4 SUBROUTINE TCHLST 9-13

9.5 SUBROUTINE HOCUTT 9-14

9.6 PROGRAM FULOI2. 9-15

9.7 SUBROUTINE GBCALC 9-17

10. FIELD DEFINITION FLDDEF. 10-1

10.1 SUBROUTINE DELFLD. 10-7

10.2 SUBROUTINE FLDRPT. 10-11

10.3 SUBROUTINE IRREG3. 10-14

10.4 SUBROUTINE CURDEF. 10-19

10.5 SUBROUTINE FAKCU3. 10-22

10.6 SUBROUTINE RDXYD3. 10-25

10.7 SUBROUTINE SDLINE. 10-28

Section Page

10.8 SUBROUTINE SDPNT 10-31

10.9 SUBROUTINE SWCLR 10-33

10.10 SUBROUTINE SORT 10-35

10.11 SUBROUTINE PAINT 10-37

10.12 SUBROUTINE FLDNAM 10-40

11. DOT GROUP CROSSHAIR OVERLAY DOTOVR 11-1

12. DOT GROUP SCATTER PLOT SCPLOT. 12-1

12.1 SUBROUTINE DGSCPL. 12-4

12.2 SUBROUTINE LRJUNK. 12-27

12.3 SUBROUTINE PLOT. 12-30

12.4 SUBROUTINE SETVID. 12-37

12.5 SUBROUTINE SETWIN. 12-40

12.6 SUBROUTINE VALCK 12-45

12.7 SUBROUTINE WINDRM. 12-48

12.8 SUBROUTINE THLOPM. 12-50

13. SINGLE DOT PROCESSING DOTPRO 13-1

13.1 SUBROUTINE STYPE 13-13

13.2 SUBROUTINE FINDOT. 13-18

13.3 SUBROUTINE PROCED. 13-32

13.4 SUBROUTINE DOTLAB. 13-34

13.5 SUBROUTINE HSEKPG. 13-44

13.6 SUBROUTINE BLOWUP. 13-46

13.7 SUBROUTINE GTYPE 13-57

Sections 13 through 22 are included in Part 2.

Section Page

13.8 SUBROUTINE TRAJPL 13-62

14. AUTOMATIC CLUSTER LABELLING PROGRAM ACLLAP 14-1

14.1 SUBROUTINE RDODAT 14-10

14.2 SUBROUTINE RDCLMN 14-14

14.3 SUBROUTINE CLABEL 14-17

14.4 SUBROUTINE ALSORT 14-21

14.5 SUBROUTINE KNNPRN 14-23

14.6 SUBROUTINE BRFCLU 14-27

14.7 SUBROUTINE REPORT 14-29

14.8 SUBROUTINE CLUSNN 14-33

14.9 SUBROUTINE REPRTN 14-35

15. CLUSTER MAP DISPLAY CLUDIS 15-1

15.1 SUBROUTINE EFWARD 15-6

15.2 SUBROUTINE UNCDIS 15-9

15.3 SUBROUTINE LIST1 15-14

15.4 SUBROUTINE CATTHM 15-17

15.5 SUBROUTINE CLUTHM 15-22

15.6 SUBROUTINE TWRITE 15-27

15.7 SUBROUTINE CLABEL 15-31

15.8 SUBROUTINE GETCOO 15-34

15.9 SUBROUTINE ZOOOOM 15-38

15.10 SUBROUTINE REPORT 15-41

15.11 SUBROUTINE DEFALT 15-43

Section Page

15.12 SUBROUTINE CONDIS 15-46

15.13 SUBROUTINE CONDIT 15-51

15.14 SUBROUTINE MIXDIS 15-54

15.15 SUBROUTINE LIST2. 15-58

15.16 SUBROUTINE MIXED. 15-61

16. CLASSIFICATION MAP DISPLAY/RECOMPUTE PROPORTIONS REPROP 16-1

16.1 SUBROUTINE FDLINT. 16-12

16.2 SUBROUTINE RECPRN. 16-16

16.3 SUBROUTINE CLADIS. 16-20

16.4 SUBROUTINE LIST3 16-25

16.5 SUBROUTINE CLATHM. 16-28

16.6 SUBROUTINE DEFLT2. 16-33

17. DOT DATA REPORT DOTRPT 17-1

18. BIAS CORRECTION/CLASSIFICATION SUMMARY BIASCR. 18-1

18.1 SUBROUTINE ROFF. 18-9

18.2 SUBROUTINE SELDOT. 18-11

18.3 SUBROUTINE UNBIAS. 18-16

18.4 SUBROUTINE ALPTAB. 18-21

19. CLUSTER REPORTS CLURPT 19-1

19.1 SUBROUTINE MENSTD. 19-13

19.2 SUBROUTINE INCLDS. 19-24

19.3 SUBROUTINE DATARD 19-39

19.4 SUBROUTINE HSEKPG 19-42

Section Page

19.5 SUBROUTINE BRFCLU 19-44

19.6 SUBROUTINE REPORT 19-46

19.7 SUBROUTINE CLUSNN 19-49

19.8 SUBROUTINE REPRTN 19-51

20. FIELD DEFINITION REPORT FLDRPT. 20-1

21. PERMANENT DATA BASE UPDATE. 21-1

21.1 SUBROUTINE AQLFST 21-6

21.2 SUBROUTINE JULIAN 21-9

21.3 SUBROUTINE DOTSAV 21-11

21.4 SUBROUTINE FLDSAV 21-14

21.5 SUBROUTINE CLASAV 21-17

21.6 SUBROUTINE STASAV 21-21

21.7 SUBROUTINE SUBSTR 21-24

21.8 SUBROUTINE RPTGEN 21-26

21.9 SUBROUTINE UNLDOT 21-31

21.10 SUBROUTINE ADDDOT. 21-35

21.11 SUBROUTINE FLDOFF. 21-37

21.12 SUBROUTINE RDFLD 21-43

22. REFERENCES. 22-1

INDEX. I-1

1. IMAGE DATA BASE UPDATE PROGRAM IMAUPD

```
HEWRTRAN IV=PLUS V02=04          12124105   29=AUG=77          PAGE 1
IMAUPD,FTN       /TR;BL@CKS/WR
0001              PR@GRAM JDIR
0002             IMPLICIT INTEGER (A=Z)
0003              L@GICAL=1 BUF
0004              L@GICAL=1 ACQNUM, DFLAG
0005              L@GICAL=1 HEADER(3060)
0006              L@GICAL=1
                C     SW(2)
                C     ,SEGJD(30)
                C     ,SEGBYT(4)
0007             L@GICAL=1
                C     DIRBYT(512)
0008             L@GICAL=4 JDATE
0009             L@GICAL=4 DIRLG4 (128)
0010              DIMENSI@N JDT(2)
0011              DIMENSI@N HEAD(23)
0012              DIMENSI@N SG(2)
0013              DIMENSI@N HDWD (1530)
0014             DIMENSI@N BUF (3072)
0015             DIMENSI@N DIAG(40)
0016              DIMENSI@N INDEXF(2,200) , DIR=RD(256)
0017             DIMENSI@N T@BD@N(6,20)
0018             DIMENSI@N SEQREC(2)
0019             DIMENSI@N DISKTL (10)
0020             DIMENSI@N JD(2)
0021              C@MM@N /AD/ ACQNUM, DFLAG
0022              C@MM@N /SKIPFL/ SKIPFL
0023              C@MM@N /NEXREC/ NREC
0024              EQUIVALENCE (JDATE, JDT)
0025              EQUIVALENCE (SG, SEGBYT)
0026              EQUIVALENCE (HEADER, HDWD)
0027             EQUIVALENCE (JD, JDATE)
0028              EQUIVALENCE (SW, SEGNUM)
0029             EQUIVALENCE (DIRBYT, DIRRED, DIRLG4)
0030              DATA NT@ /0/, BUFSZ /3072/
0031              DATA KDX/200/
0032              DATA DATA /0/
0033              DATA MIMTFG /1/
0034              DATA KD@TS /209/
0035              DATA  MAXACD/6/, DLSKIP/10/
0036              DATA MAXCHN/4/
0037              DATA MAXL@D /80/
0038              DATA SEGJD /1HD, 1HB, 1H2,1H1, 1HC, 1H3, 1H0,
                C     1H@, 1H,, 1H3, 1H0, 1H@, 1H3, 1H ,1H ,
                C     1H , 1H ,1H , 1H : 1H , 1H ,1H ,
                C     1H.,  1HD, 1HA, 1HT, 1H , 1H , 1H , 1H /
0039              TF@RM = 1
0040              UNIT = 0
0041              F11 = 1
0042              ENDFIL = 0
0043             KI = KDX * 2
0044              @PEN (UNIT=3,NAME='[300,300]@SKTBL.DAT;1',
                C     TYPE='@LD', ACCESS='DIRECT',ASS@CIATEVARIABLE=@REC,
                C     MAXREC=400, REC@RDSIZE=200)
0045              @PEN (UNIT=4, NAME='DB21[300,300]HDRREC,DAT',
                C     ACCESS='SEQUENTIAL', TYPE ='@LD', READ@NLY)
0046             READ (3'1) INDEXF
```

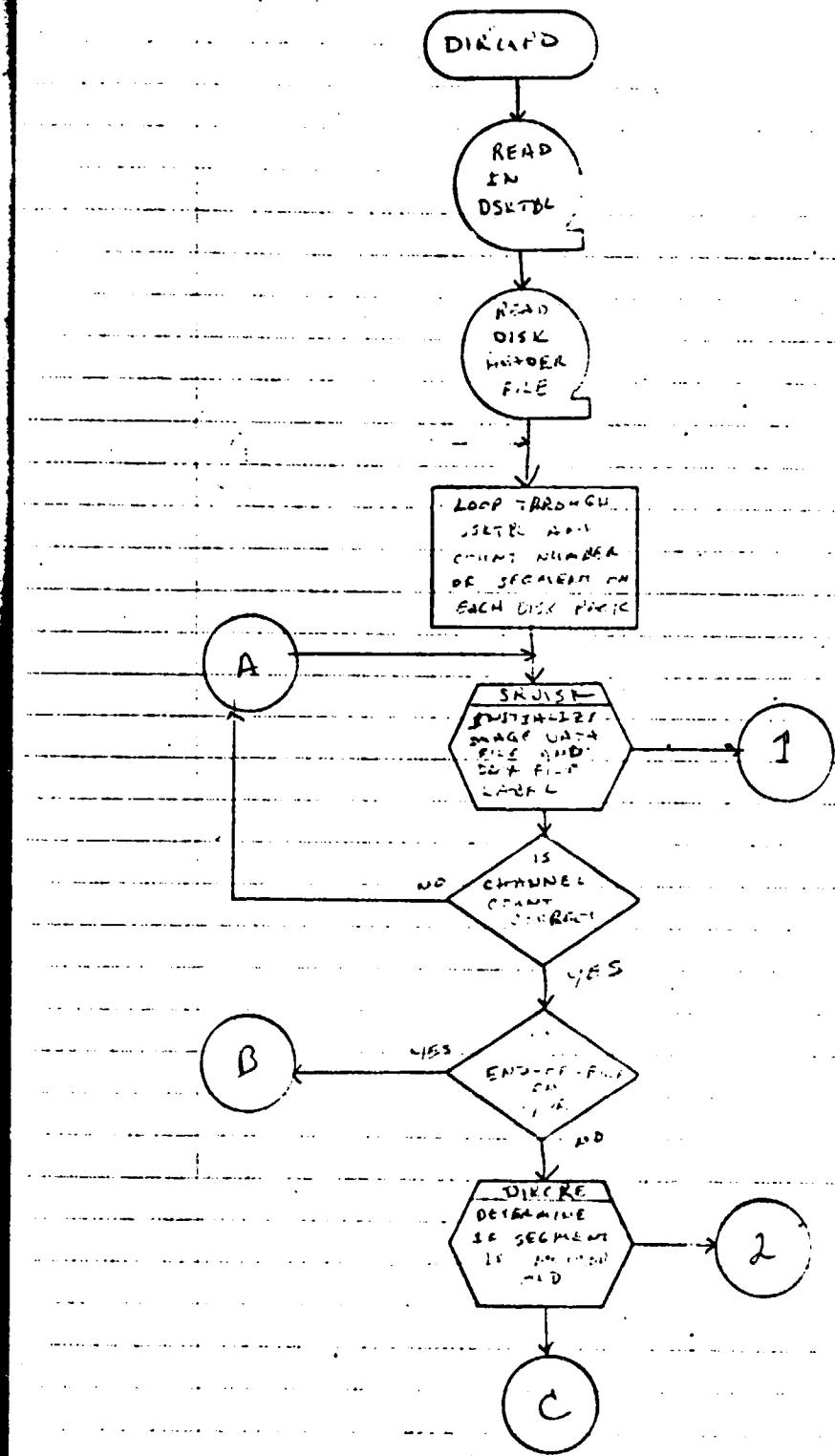
```
0047              READ (4,90) HEAD
0048              ØPEN(UNIT=13,NAME='DIAGN.DAT;1',TYPE='SCRATCH')
0049           90 FØRMAT(23A2)
0050              DECØDE (4, 100, HEAD) DSKNUM
0051          100 FØRMAT(I4)
0052              DØ 1400 NI = 1, KDX
0053              IND = INDEXF (2, NI)
0054              IF (INDEXF(1,NI) .GT. 0) DISKTL(IND) = DISKTL(IND) + 1
0055         1400 CØNTINUE
0056              ØPEN (UNIT=11, NAME='REPØRT.DAT', TYPE='SCRATCH',
                C     FØRM='UNFØRMATTED')
0057              IF (IFØRM - 2) 1500, 1500, 1500
0058         1450 CØNTINUE
0059              CLØSE (UNIT = 2)
0060         1500 CØNTINUE
0061              ØPEN(UNIT=2,NAME='[300,300]DIRFILE.DAT;1',TYPE='ØLD',
                C     ACCESS='DIRECT',ASSØCIATEVARIABLE=NREC,
                C     MAXREC=200, RECØRDSIZE=128)
0062              LUNIMA = 1
0063              LUD = 7
0064              IØØ = 0
0065              ØYR = 0
0066              ØDAY = 0
                C*
                C*    READ HEADER FRØM TAPE, UNPACK SEGMENT NUMBER AND ACQUISITIØN
                C*    DATE, BUILD DATA BASE FILE NAME FØR IMAGE DATA IN 'SEGJD'
                C*    ARRAY.  MAKE SURE IMAGE HAS CØRRECT NUMBER ØF CHANNELS.
                C*
0067              CALL SRDISK
                C     (LUNIMA, IFØRM, JDATE,
                C     SEGTYT,
                C     SEGJD, HEADER, BUF, BUFSZ, LUD
                C     ,SW, SEGNUM, IYR, JDAY
                C     ,DSKNUM, ENDFIL
                C     ,MAXCHN
                C     ,MXCHAN, HDWD, JDT, F11
                C     ,MTNTFG, UNIT
                C     )
0068              IF ( MXCHAN .GT. 0) GØ TØ 1450
0069              IF (ENDFIL .EQ. 1) GØ TØ 6000
                C*    CHECK THE DISK TABLE FILE TØ LØCATE THE SEGMENT IN THE
                C*    DIRECTØRY AND MAKE SURE THE CØRRECT DISK PACK IS MØUNTED,
                C*    IF NEW SEGMENT, DETERMINE DIRECTØRY INDEX (DRLINE) RECØRD,
                C*
0070              CALL DIRCRE
                C     (SEGNUM,  JDATE,
                C      INDEXF
                C     ,DSKNUM, TØRDEN, NTØ, NZGV
                C     ,KDX, DISKTL, JD
                C     ,MAXLØD, DRLINE, DIRWØD
                C     ,LUNIMA, F11
                C     )
0071              IF (NØGØ .EQ. 1) GØ TØ 1450
                C*
                C*    READ DIRECTØRY FØR THIS SEGMENT, CHECK NUMBER ØF ACQUISITIØNS,
                C*    AND DELETE FLAG.  IF ACQUISITIØN CANNØT BE ADDED, SKIP FILE
```

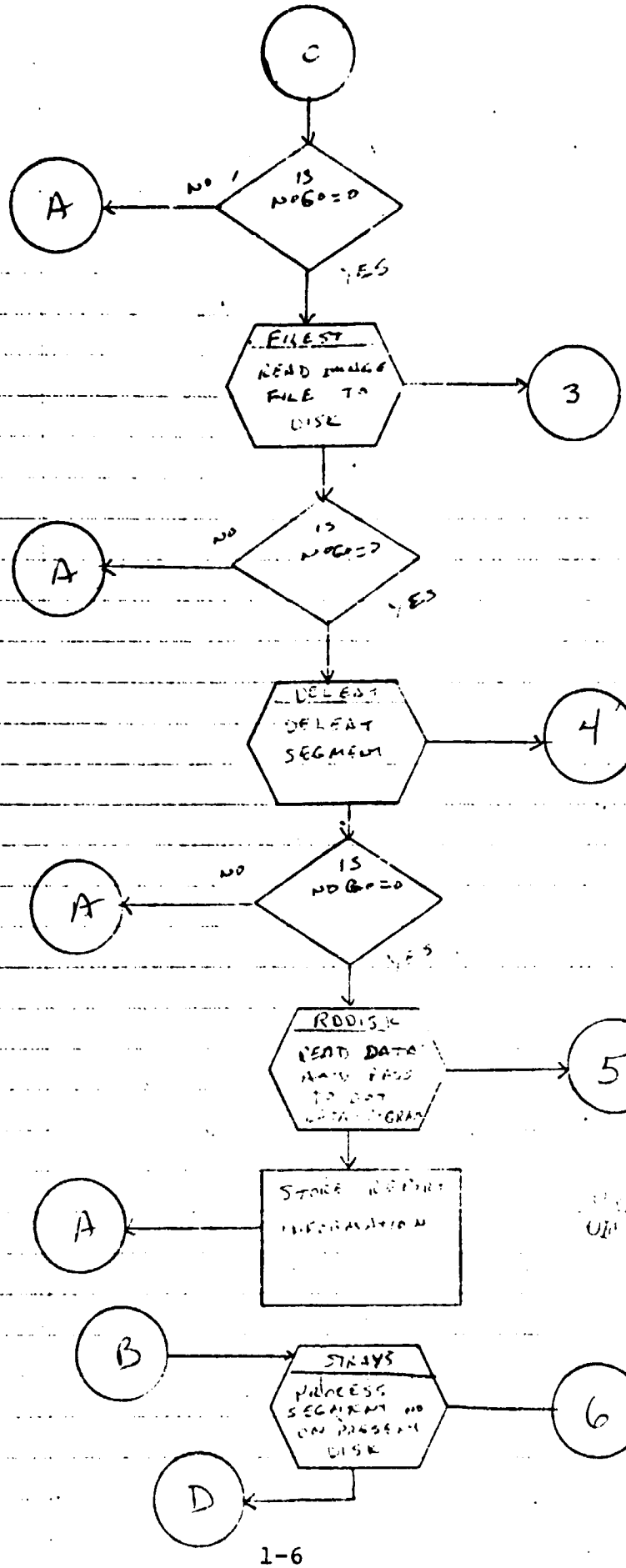

ORIGINAL PAGE IS
OF POOR QUALITY

```
        C*      ØN TAPE AND SET NØGØ=1, PERFØRM ACQ. DELETIØN IF REQUIRED,
        C*
0072            CALL DELEAT
              C     (SEGNUM, JDATE , DIRBYT,
              C     DIRLG4, DIRWRD, DRLINE, HEADER
              C     ,MAXACD, HDWD
              C     ,DYR, DDAY
              C     ,NØGØ
              C     ,GREEN, ANG, AZM
              C     ,JD
              C     )
0073            IF(NØGØ.GE.1)SKIPFL=1
        C*
        C*      READ THE IMAGE FRØM TAPE AND STØRE ØN DISK
        C*
0074            IF(NØGØ.GE.1)CALL DUSET(LUNIMA,F11,1)
0075            IF (NØGØ .GE. 1) GØ TØ 1450
0076            CALL FILEST
              C     (LUNIMA,LUD,TFØRM,BUF,BUFSZ,SEGJD
              C     ,TØBDØN,NTØ,MTMTFG
              C     ,UNIT,NRECR,NPRTYR,NRLØCK
              C     ,NØGØ
              C     ,F11,SEGNUM
              C     )
0077            IF (NØGØ.EQ.1)GØ TØ 1450
        C*
        C*      READ THE IMAGE DATA FRØM THE DISK FILE, BUILD ØR UPDATE THE
        C*      DØT DATA FILE, UPDATE THE DIRECTØRY FILE.
        C*
0078            CALL RDDISK
              C     ( LUD, JDATE , SEGJD, TFØRM,
              C     HEADER, BUF, BUFSZ
              C     ,DIRWRD, DIRBYT, DIRLG4
              C     ,KDX, DLSKIP, MAXCHN
              C     ,MAXACD, NDØTS
              C     ,SEGNUM
              C     ,HDWD, DRLINE
              C     )
0079            WRITE (3'1) INDEXF
0080            AQ =ACQNUM
0081            DFG = DFLAG
0082            WRITE (11) SEGNUM, IYR, JDAY, DYR, DDAY,
              C     NRECR, NPRTYR, NRLØCK, AQ, DFG
              C     ,GREEN, ANG, AZM
0083            KLINE = KLINE + 1
0084            GØ TØ 1450
0085     6000   CØNTINUE
0086            WRITE (3'1) INDEXF
0087            CALL CLØSE (3)
        C*
        C*      REWIND THE TAPE
        C*
0088            CALL TRWD(LUNIMA)
0089            IF(KLINE.EQ.0)GØ TØ 705
0090            REWIND 11
0091            DØ 7000 KL = 1, KLINE
```

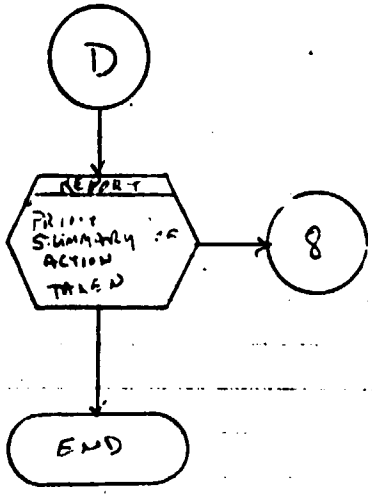
ORIGINAL PAGE IS
OF POOR QUALITY

```
0092              READ (11) SEGNUM , IYR, JDAY, DYR, DDAY,
                C       NRECR, NPRTYR, NBLØCK, AQ, DFG
                C      ,GREEN, ANG, AZM
0093              CALL REPØRT (SEGNUM,  IYR, JDAY, DYR, DDAY,
                C       NRECR, NPRTYR, NBLØCK,
                C       AQ, DFG, GREEN, ANG, AZM)
0094       7000   CØNTINUE
0095      705     CØNTINUE
0096              WRITE(6,700)
0097              REWIND 13
0098               DØ 7001 KL=1,1000
0099              READ(13,701,END=7002)DIAG
0100              WRITE(6,701)DIAG
0101      701     FØRMAT(60A2)
0102      7001    CØNTINUE
0103      7002    CØNTINUE
0104              WRITE (6,700)
0105        700   FØRMAT(1HØ)
0106              CALL DUSET (LUNIMA, F11, 0)
0107              STØP 1
0108              END
```


DIRLOAD
READ IN DSKTBL
READ DISK DIRECTORY FILE
LOOP THROUGH DSKTBL AND COUNT NUMBER OF SEGMENTS ON EACH DISK PACK
A
SRJISR
INITIALIZE IMAGE DATA FILE AND OUT FILE LABEL
1
IS CHANNEL COUNT ZERO?
NO
YES
B
YES
END-OF-FILE ON DIR.
NO
DIRCRE
DETERMINE IF SEGMENT IS DIRECTORY
2
C



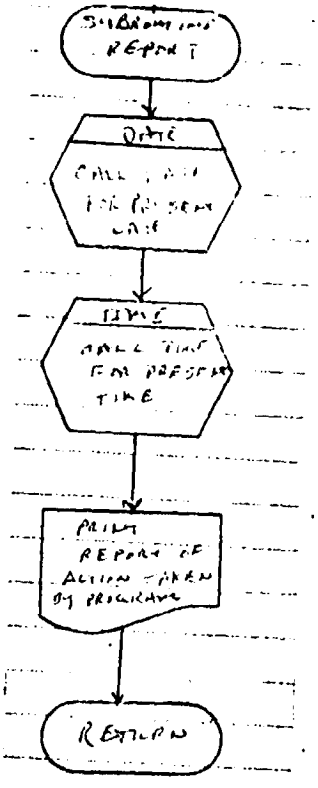
ORIGINAL PAGE IS
OF POOR QUALITY



ORIGINAL PAGE IS
OF POOR QUALITY

1.1 SUBROUTINE REPORT

```
FORTRAN IV-PLUS V02-04          12124116    29-AUG-77          PAGE 7
IMAUPD,FTN      /TR:BLOCKS/WR
0001            SUBROUTINE REPORT
                C     (SEGNUM,  IYR, JDAY, DYR, DDAY,
                C      NREC#, NPRTY#, NBLOCK,
                C     A#, DFG, GREEN, ANG, AZM)
0002            IMPLICIT INTEGER (A-Z)
0003            LOGICAL*1 KDATE, KTIME(8)
0004            COMMON /DATE/ KDATE(9)
0005            DATA LINES /0/, PAGE /0/
0006            IF (LINES .GT. 0) GO TO 2000
0007            IF (PAGE .GT. 0) GO TO 1100
0008            CALL DATE (KDATE)
0009            CALL TIME (KTIME)
0010      1100  CONTINUE
0011            PAGE = PAGE + 1
0012            WRITE (6, 120) PAGE, KDATE
0013       120  FORMAT (1H1,
                C     T44, 'CAMS-1100 DATA BASE '
                C     'TRANSACTION REPORT',
                C     T105, 'PAGE ',I3,
                C     /,T104, 'DATE: ', 9A1)
0014            WRITE (6, 140)  KTIME
0015       140  FORMAT (1X, T44,
                C     'IMAGE DATA BASE UPDATE / '
                C     ,'DAY', 1X, '15', ' VERSION',
                C     T105, 'TIME: ', 8A1,/)
0016            WRITE (6,160)
0017      160   FORMAT (1X,
                C     T13,'ACQ.', T20,'NUMBER', I38,'ACQ.', T70,'NUMBER',
                C     T82,'NUMBER',/,
                C     1X,T13,'DATE', T22,'YR', T30,'ACQ.', T38,'DATE',
                C     T47,'SUN', T55,'SUN', T62,'GREEN', T70,'OF IMAGE',
                C     T80,'PARITY', T89,'NUMBER',/,
                C      2X,'SEGMENT', T12,'ADDED', T21,'ACQ.' T28,'DELETED',
                C     T37,'DELETE',
                C     T46,'ANGLE', T53,'AZIMUTH', T62,'NUMBER', T71,'RECORD',
                C     T82,'RECORDS', T89, 'BLOCK(S)')
0018      200   CONTINUE
0019            WRITE(6, 200)
                C     SEGNUM, IYR, JDAY, A#, DFG, DYR, DDAY, ANG,
                C     AZM, GREEN, NREC#,  PRTY#, NBLOCK
0020       220  FORMAT (1X, T5, I4, T12,I2, 1X, I3, T21,I2, T31,I2,
                C     T38,I2, 1X, I3, T47,I3, T55,I4, T64,I2, T73,I3,
                C     T82,I3, T91,I8)
0021            LINES = LINES + 1
0022            IF (LINES .GE. 58) LINES = 0
0023            RETURN
0024            END
```

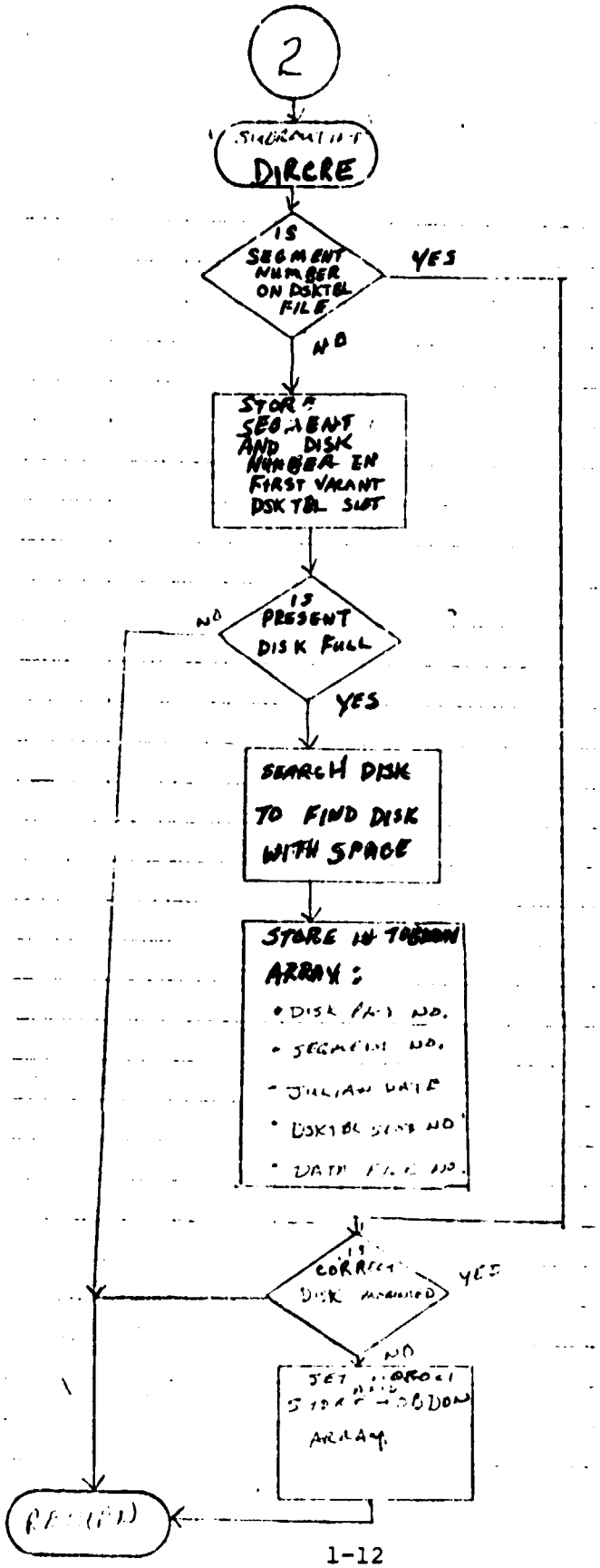

REPORT
DATE
TIME
REPORT
RETURN

1.2 SUBROUTINE DIRCRE

```
FØRTRAN IV-PLUS V02-04          12:24:20    29-AUG-77          PAGE 9
IMAUPD.FTN      /TRIBLOCKS/WR
0001            SUBROUTINE DIRCRE
                C     (SEGNUM,  JDATE ,  INDEXF
                C     ,DSKNUM, TØRDØN, NTØ, NØG
                C     ,KDX    , DISKTL, JD
                C     ,MAXLØD, DRLINE, DIRWRD
                C     ,LUNIMA, F11
                C     )
0002            IMPLICIT INTEGER (A-Z)
0003            LØGICAL*4 JDATE
0004            DIMENSIØN TØRDØN(6,20)
0005             DIMENSIØN DIRWRL(256)
0006            DIMENSIØN DISKTL(10)
0007            DIMENSIØN JD(2)
0008            DIMENSIØN INDEXF(2, KDX)
0009             CØMMØN /SKIPFL/ SKIPFL
0010            IMPTY = 0
0011            NØG = 0
0012             NPACK = DSKNUM
0013             KDX = 200
0014             DØ 2700 NS = 1, KDX
0015             DRLINE = NS
0016            IF (INDEXF(1,NS) .LE. 0 .AND. IMPTY .EQ. 0) IMPTY = NS
0017            IF (INDEXF(1,NS) .EQ. SEGNUM) GØ TØ 3000
0018     2700   CØNTINUE
0019            IF (IMPTY .EQ.0) GØ TØ 9000
0020             DRLINE = IMPTY
0021            INDEXF(1,IMPTY) = SEGNUM
0022             INDEXF(2,IMPTY) = NPACK
0023            IF (DISKTL(NPACK) .LT. MAXLØD)  GØ TØ 2775
0024            DØ 2765 LH = 1, 10
0025            NPACK = LH
0026            IF (DISKTL(LH) .LT. MAXLØD) GØ TØ 2770
0027     2765   CØNTINUE
0028     2770   CØNTINUE
0029            NTØ = NTØ + 1
0030             IF(NTØ.GT.20)NTØ=20
0031            DISKTL(NPACK) = DISKTL(NPACK) + 1
0032            TØRDØN (1,NTØ) =  PACK
0033            TØRDØN (2,NTØ) = SEGNUM
0034            TØRDØN (3,NTØ) = JD(1)
0035            TØRDØN (4,NTØ) = JD(2)
0036            TØRDØN (5, LH) = IMPTY
0037     2775   CØNTINUE
0038     2800   CØNTINUE
0039            GØ TØ 4000
0040     3000   CØNTINUE
0041            NPACK = INDEXF(2,DRLINE)
0042            IF (NPACK .EQ. DSKNUM) GØ TØ 4000
0043            NØG = 1
0044            NTØ = NTØ + 1
0045             IF(NTØ.GT.20)NTØ=20
0046            TØRDØN(1, NTØ) = NPACK
0047            TØRDØN (2, NTØ) = SEGNUM
0048            TØRDØN(3,NTØ) = JD(1)
0049            TØRDØN (4,NTØ) = JD(2)
0050            TØRDØN(5,NTØ) = DRLINE
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
FØRTRAN IV-PLUS V02-04          12:24:20    29-AUG-77          PAGE 10
IMAUPD.FTN      /TR:BL0CKS/WR
0051            WRITE(13,300) SEGNUM, DSKNUM, NPACK
0052       300  FØRMAT (1HØ,
                C     ' SEGMENT NUMBER ' ,I6, ' NØT FØUND ØN '
                C     'THIS DISK PACK, ' ,I3,/,
                C     '  THIS SEGMENT IS ØN DISK PACK ', I3)
0053            CALL DUSET(LUNIMA, F11, 1)
0054            SKIPFL = 1
0055       4000 CØNTINUE
0056            RETURN
0057       9000 CØNTINUE
0058            WRITE(13,900) SEGNUM
0059        900 FØRMAT (1HØ,
                C     '  THE DATA DIRECTØRY FILE IS FULL.', /,
                C     '   SEGMENT NUMBER ', I5, ' CANNØT BE ADDED TØ THAT FILE')
0060            NØG0=1
0061            SKIPFL=1
0062            CALL DUSET(LUNIMA,F11,1)
0063            RETURN
0064            END
```


2
DIRCRE
IS SEGMENT NUMBER ON DSKTBL FILE
YES
NO
STORE SEGMENT AND DISK NUMBER IN FIRST VACANT DSK TBL SLOT
IS PRESENT DISK FULL
NO
YES
SEARCH DISK TO FIND DISK WITH SPACE
STORE IN TABLE ARRAY:
• DISK PACK NO.
• SEGMENT NO.
• JULIAN DATE
• DSKTBL SLOT NO
• DATA FILE NO.
IS CORRECT DISK MOUNTED
YES
NO
SET ERROR STORE IN TABLE ARRAY

1.9 SUBROUTINE DELEAT

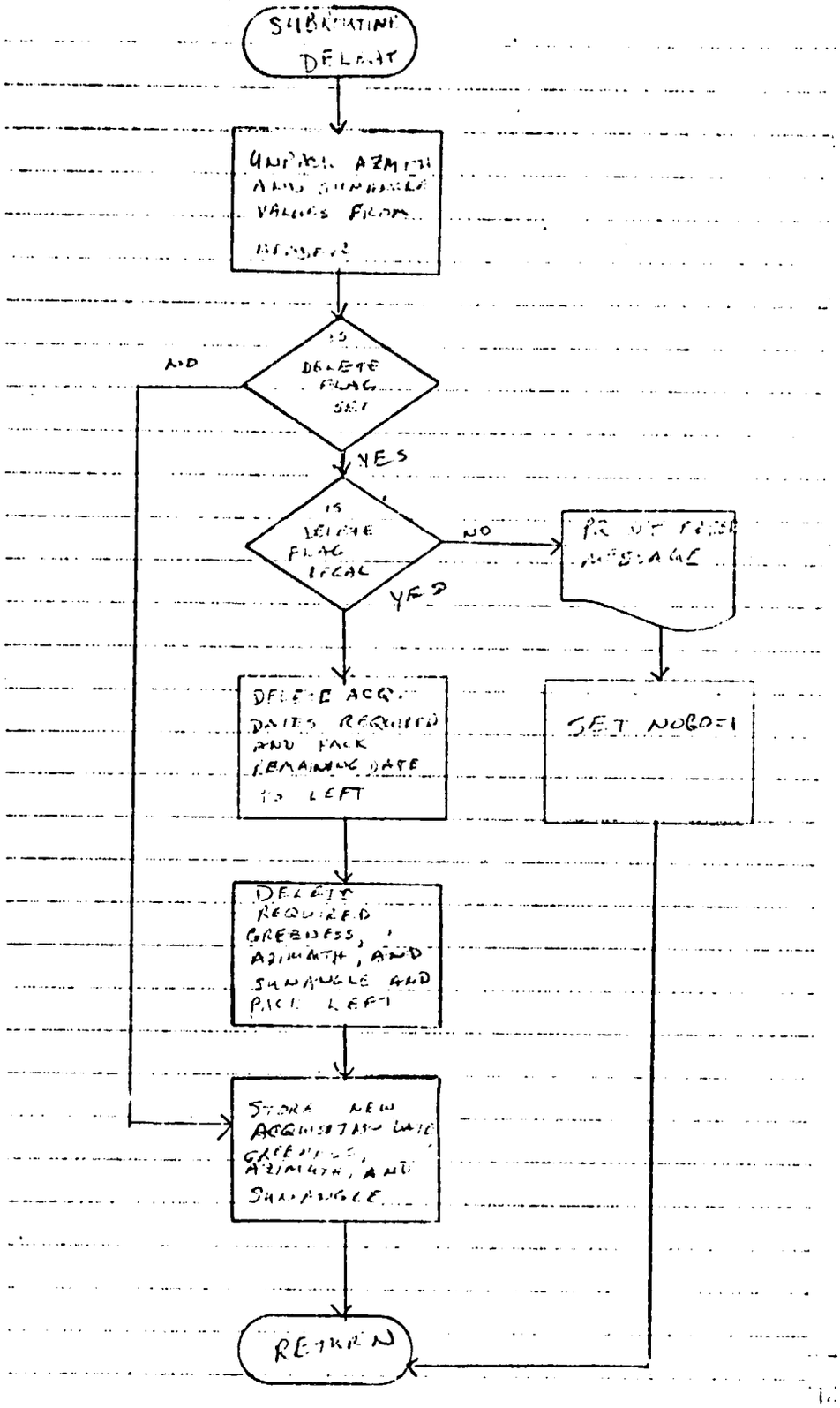
```
FØRTRAN IV+PLUS V02-04          12:24:27    29-AUG-77          PAGE 18
IMAUPD.FTN      /TR:BLØCKS/HR
0001            SUBRØUTINE DELEAT
                C     (SEGNUM , JDATE , DIRBYT,
                C     DIRLG4, DIRWRD, DRLINE, HEADER
                C     ,MAXACD, HDWD
                C     ,DYR, DDAY
                C     ,NØGØ
                C     ,GREEN, ANG, AZM
                C     ,JD
                C     )
0002              IMPLICIT INTEGER (A-Z)
0003              LØGICAL*1 DIRBYT(512)
0004              LØGICAL*1 SEGDEL(30)
0005              LØGICAL*1 ACONUM, DFLAG
0006              LØGICAL*4 JDATE,KDATE
0007              LØGICAL*4 DIRLG4 (128)
0008              LØGICAL*1 HEADER(3060)
0009              DIMENSIØN DIRWRD(256)
0010              DIMENSIØN KDAY(2)
0011              DIMENSIØN JD(2)
0012              DIMENSIØN HDWD (1530)
0013              CØMMØN /AD/ ACONUM, DFLAG
0014              CØMMØN /NEXREC/ NREC
0015              EQUIVALENCE (KDATE,KDAY)
0016              DATA SEGDEL /1HD, 1HR, 1H2, 1H1,
                C     1HC, 1H3, 1H0, 1H0, 1H,, 1H3, 1H0, 1H0, 1H3,
                C     1H , 1H , 1H ,  1H ,
                C     1H , 1H , 1H , 1H , 1H ,
                C     1H,, 1HD, 1HA, 1HT, 1H), 1H1, 0,0/
0017              KDATE=JDATE
0018              FIND (2'DRLINE)
0019              READ (2'NREC) DIRWRD
0020              IF (DIRWRD(1) .LE. 0) DIRWRD(1) = SEGNUM
0021              IF (DIRBYT(4) .GE. 0) GØ TØ 1100
0022              DIRBYT(4) = 0
0023        1100  CØNTINUE
0024              ACONUM = DIRBYT(1)
0025              IF (DIRBYT(3) .GE. 0) GØ TØ 1150
0026              DIRBYT (3) = 0
0027        1150  CØNTINUE
0028              DØ 10 I=1,ACONUM
0029        10      IF(JDATE.EQ.DIRLG4(I+1))GØ TØ 9200
0030              DFLAG = DIRBYT(3)
0031              GREEN = HEADER(2254)
0032              ANG = HDWD(1101)
0033              CALL SWAB (ANG)
0034              AZ1 = HEADER(2170)
0035              AZ1 = IAND (AZ1,"377)
0036              AZ1 = AZ1 - 240
0037              AZ2 = HEADER(2169)
0038              AZ2 = IAND (AZ2, "377)
0039              AZ2 = AZ2 - 240
0040              AZ3 = HEADER(2168)
0041              AZ3 = IAND( AZ3, "377)
0042              AZ3 = AZ3 - 240
0043              AZM = (AZ3 * 100) + ( AZ2 * 10) + AZ1
0044              NAQ = ACONUM
```

```
0045            IF (DFLAG .EQ. 0 .AND. NAQ .GE. MAXACQ) GO TO 9000
0046          * IF (DFLAG .GT. MAXACQ) GO TO 9100
0047            IF (DFLAG .EQ. 0) GO TO 1500
0048            IF (NAQ .GE. DFLAG) GO TO 1350
0049             DIRBYT(3) = 0
0050             DFLAG = 0
0051            WRITE(13,130) SEGNUM, DFLAG, NAQ
0052      130   FORMAT (1H0,
                C       ' THE DELETE FLAG FOR SEGMENT NUMBER ',
                C       I5, ' IS SET TO ',
                C       I2,/,
              * C       ' THERE ARE ONLY ', I2, ' ACQUISITIONS STORED '
                C       ,/, ' THE NEW ACQUISITION WILL BE ADDED '
                C       ' AND NO DELETION WILL TAKE PLACE')
0053             GO TO 1500
0054      1350  CONTINUE
0055            IF ( NAQ .EQ. DFLAG) GO TO 1400
0056            NAQ = NAQ - 1
0057            DO 4000  JJ = DFLAG, NAQ
0058            LD = JJ + 1
0059            LG =((MAXACQ + 1) * 4) + JJ
0060            LA = LG + MAXACQ
0061            LJ = (MAXACQ * 3) + 2 + JJ
0062            MD = LD + 1
0063            MG = LG + 1
0064            MA = LA + 1
0065            MJ = LJ + 1
0066             IF (JJ .NE. DFLAG) GO TO 1375
0067             KD = LD * 2 - 1
0068             KA = LA * 1
0069             DYR = DIRWRD(KD)
0070             DDAY = DIRWRD(KA)
0071      1375  CONTINUE
0072            DIRLG4(LD) = DIRLG4(MD)
0073            DIRBYT(LG) = DIRBYT(MG)
0074            DIRBYT(LA) = DIRBYT(MA)
0075            DIRWRD (LJ) = DIRWRD(MJ)
0076      4000  CONTINUE
0077      1500  CONTINUE
0078            NAQ = NAQ + 1
0079      1400  CONTINUE
0080             IF (DFLAG .EQ. 0) GO TO 1650
0081             IF (DFLAG .NE. MAXACQ) GO TO 1651
0082             KA = ACQNUM * 2 + 1
0083             DYR = DIRWRD(KA)
0084             DDAY = DIRWRD(KA+1)
0085      1450  CONTINUE
0086            AD = NAQ + 1
0087            NG = ((MAXACQ + 1) * 4) + NAQ
0088            NA = NG + MAXACQ
0089            NJ = (MAXACQ * 3) + 2*NAQ
0090            DIRLG4(ND) = JDATE
0091             DIRWRD(36) = JD(1)
0092             DIRWRD(37) = JD(2)
0093            DIRBYT(NG) = GHEE
0094            DIRBYT(NA) = ANG
```

THIS PAGE IS
BEST QUALITY

```
0095            DIRWRD(N2) = A2M
0096             IF (DYR .EQ. 0) G0 T0 2000
0097             ENC0DE (2, 170, SEGDEL(18)) DYR
0098             ENC0DE (3, 175, SEGDEL(20)) DDAY
0099             ENC0DE (4, 180, SEGDEL(14)) SEGNUM
0100       170   F0RMAT(I2)
0101       175   F0RMAT(I3)
0102       180   F0RMAT(I4)
0103             D0 1900 FN = 14, 22
0104             IF (SEGDEL(FN) .EQ. ' ') SEGDEL(FN) = '0'
0105      1900   CONTINUE
0106             OPEN (UNIT=12, NAME=SEGDEL, ACCESS='DIRECT',
                C      TYPE='0LD')
0107             CL0SE (UNIT=12,DISP0SE='DELETE')
0108      2000   CONTINUE
0109             RETURN
0110      9000   CONTINUE
0111             WRITE(13,900) SEGNUM, ACQNUM
0112       900   F0RMAT (1H0,
                C      'ERR0R IN DELEAT =',/,
                C      ' THE DIRECTORY FILE F0R SEGMENT NUMBER ', I4,
                C      ' D0ES N0T C0NTAIN A DELETE FLAG', /,
                C      ' THERE ARE ', I3, 'ACQUISITI0NS')
0113             WRITE(13,901)KDAY
0114      901    F0RMAT(/' ACQUSITI0N',2I4,' BY-PASSED'/)
0115             G0 T0 9999
0116      9100   CONTINUE
0117             DFL = DFLAG
0118             WRITE(13,910) SEGNUM, DFL,KDAY
0119       910   F0RMAT (1H0,
                C      ' THE DFLETE FLAG F0R SEGMENT NUMBER', I5,
                C      ' IS EQUAL T0 ', I2,/,
                C      '  THE DELETE FLAG CAN RANGE FR0M 0-6',/,
                C      '  THE PR0GRAM WILL N0T PR0CESS ACQUISITI0N',2I4)
0120             G0 T0 9999
0121      9200   CONTINUE
0122             WRITE(13,920)KDAY,SEGNUM
0123      920    F0RMAT(/'  ACQUISITI0N',2I4,' F0R SEGMENT',I5,
                C  ' IS ALREADY IN THE DATA BASE---FILE BY-PASSED')
0124      9999   CONTINUE
0125             N0G0 = 1
0126             END
```

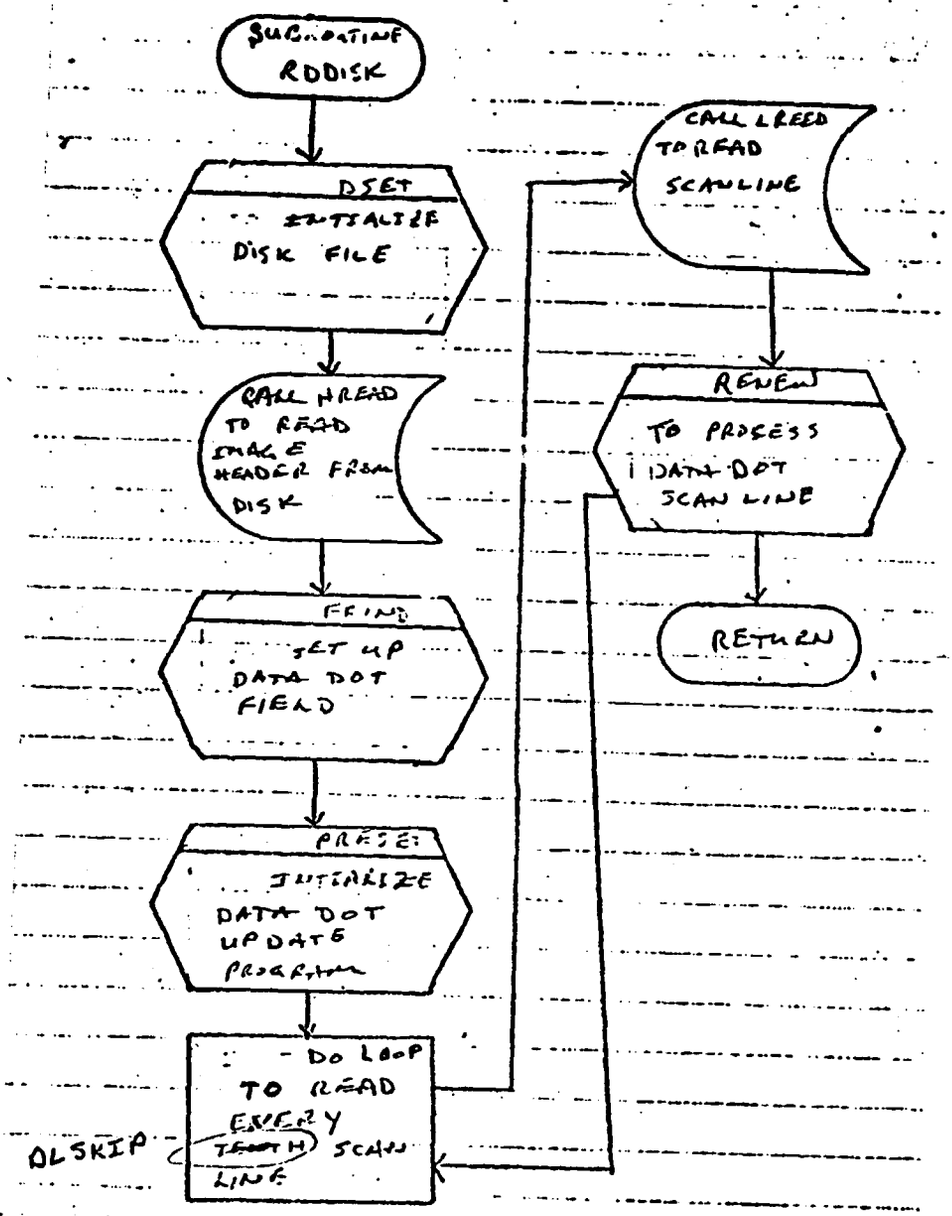
POOR QUALITY



1.4 SUBROUTINE RDDISK

```
FØRTRAN IV+PLUS V02-04          12:24:41    29-AUG-77          PAGE 17
IMAUPD.FTN      /TR:BLØCKS/WR
0001            SUBRØUTINE RDDISK
                C     (LUD,  JDATE , SEGJD, TFØRM,
                C     HEADER, BUF, BUFSZ
                C                  , DIRWRD, DIRBYT,DIRLG4
                C     ,KDX, DLSKIP, MAXCHN
                C     ,MAXACD, NDØTS
                C     ,SEGNUM
                C     ,HDWD, DRLINE
                C     )
0002               IMPLICIT INTEGER (A-Z)
0003               LØGICAL*1
                C     BUF(BUFSZ), HEADER(3060),
                C     IDATA (196,4)
                C    ,DIRBYT(512)
                C     ,SEGJD(30)
0004               LØGICAL*1 DDFILE (29)
0005             LØGICAL*4 JDATE
0006             LØGICAL*4 DIRLG4 (128)
0007               DIMENSIØN DIRWRD(256)
0008               DIMENSIØN HDWD(1530)
0009               DIMENSIØN SG(2)
0010               CØMMØN /IØLP/ IO, LP
0011               CØMMØN /NXRR/ NXREC
0012               CØMMØN /NEXREC/ NREC
0013               CØMMØN /AD/ ACQNUM, DFLAG
0014               CØMMØN /STATUS/ S(2)
0015             CØMMØN /HCØM/
                C     SS, SE, LS, LE, NRPDS, NDSPR,
                C     NCPR, NRPC, ANCL, NC, NS,
                C     NBIT, DPI, NCAR, SVD, RSIZ,
                C     PSKIP, HSIZ, CALP, CERR
0016               DATA ØLDSEG /0/
0017               DATA IO /5/, LP /6/
0018               DATA DDFILE /1HD, 1HR, 1H2, 1H:, 1HC, 1H3, 1HØ,
                C     1HØ, 1H,, 1H3, 1HC, 1HØ, 1H3, 1H , 1H ,
                C     1H , 1H , 1HD, 1HD, 1H2, 1HT, 1HS, 1H,, 1HD,
                C     1HA, 1HT, 1H:, 1H:, 1H /
0019             F11 = 0
0020               LUD = 7
0021             SHFG = 1
0022               ENCØDE (4, 100, DDFILE(14)) SEGNUM
0023       100   FØRMAT (I4)
0024               DØ 101 I=14,17
0025               IF(DDFILE(I) .EQ. ' ') DDFILE(I)='0'
0026       101   CØNTINUE
0027               ØPEN (UNIT=10, NAME=DDFILE, TYPE='UNKNØWN',
                C     ACCESS='DIRECT',MAXREC=NDØTS, RECØRDSIZE=13,
                C     ASSØCIATEVARIABLE=NXREC)
0028               CALL DSET (LUD, 0, SEGJD, MTXTFG, UNIT, FN, 0)
0029             CALL HREAD
                C     (LUD, HEADER,, BUF, 3072,
                C     TFØRM, 0, PRTY, EØF, SHFG)
0030               IF (CERR .GE. 1) GØ TØ 9100
0031             CALL FFIND
                C     (LUD, SS, LS, SE, LE,
                C     TFØRM, BUF, BUFSZ)
```

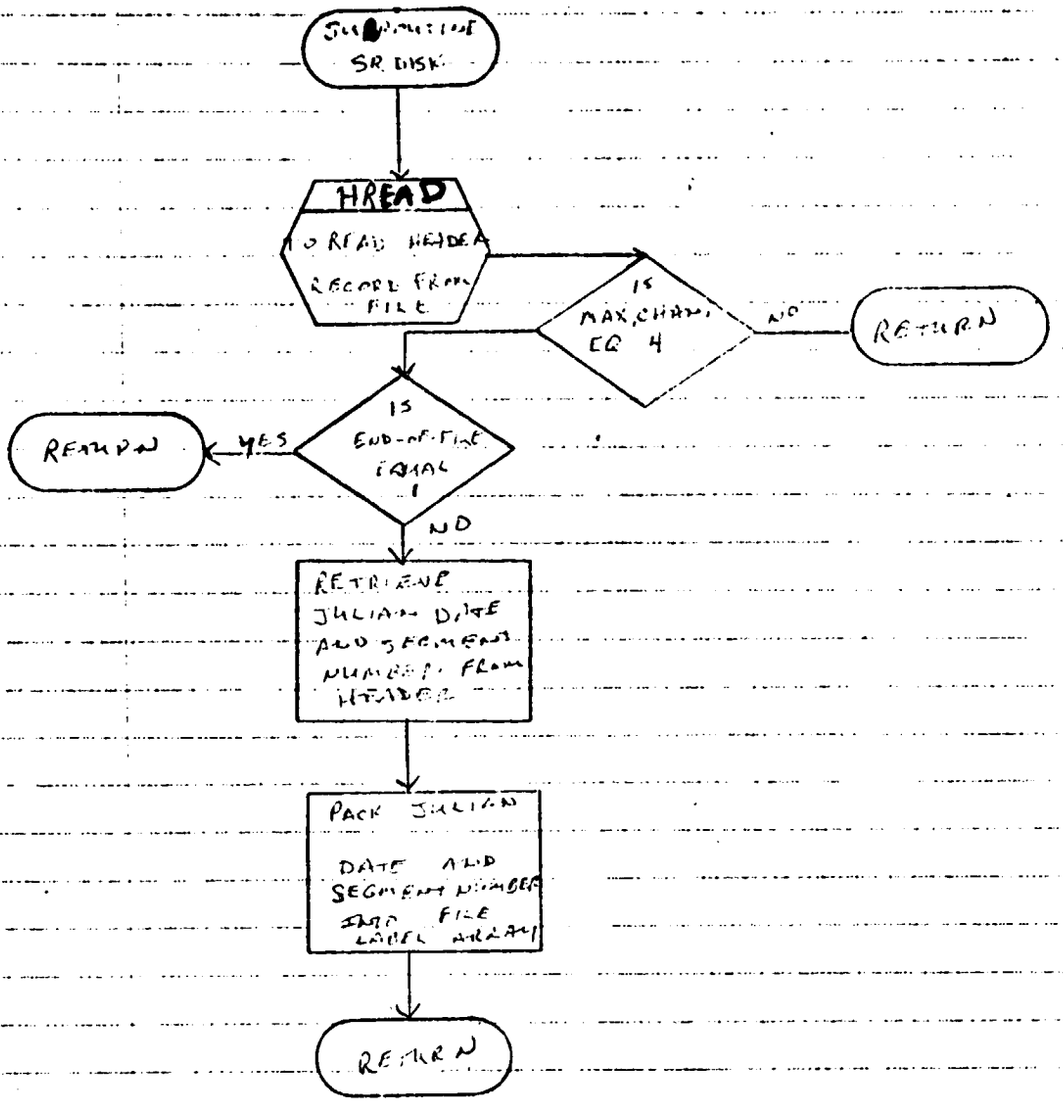
```
IMAUPD,FT4      /IR:PLØCKS/WR
0032            IF (CERR .GE. 1) GØ TØ 9200
0033           CALL PRESET
               C     (DIRWRD, DIRBYT, DIRLG4)
0034           DØ 3000 I = DLSKIP, 117, DLSKIP
0035            DØ 2000 IC = 1, MAXCHN
0036            CALL LREED
               C     (LUD, IDATA(1,IC), BUF, BUFSZ,
               C     I, F11, PRTY, IERF, TERRM, IC)
0037            IF (CERR .GE. 1) GØ TØ 9300
0038     2000   CØNTINUE
0039            CALL RENEW (IDATA, DDFILE)
0040     3000   CØNTINUE
0041            CALL CLØSE (10)
0042            IF (DFLAG .GT. 0 .AND. DFLAG .LE. ACQNUM) GØ TØ 3500
0043           IF (ACQNUM .LT. MAXACQ) ACQNUM = ACQNUM + 1
0044     3500   CØNTINUE
0045            DIRBYT(4) = ACQNUM
0046            DIRBYT(3) = 0
0047            ØLDSEG = SEGNUM
0048            CALL FVCLØS (LUD)
0049            CLØSE (UNIT = 2)
0050            ØPEN(UNIT=2, NAME='[300,60]DIRFILE.DAT;1',
               C     TYPE='ØLD', ACCESS='DIRECT',
               C     ASSØCIATEVARIABLE=NREC, MAXREC=200)
0051            FIND (2'DRLINE)
0052            WRITE(2'NREC)DIRWRD
0053            RETURN
0054     9100   CØNTINUE
0055            WRITE (13, 910) LUD
0056      910   FØRMAT (1HØ,
               C     '*ØDISK ERRØR - A FATAL ERRØR WAS RETURNED BY HREAD '
               C     'WHILE TRYING TØ READ',/,
               C     ' THE HEADER RECØRD ØF THE IMAGE TAPE.  LUN = ', I2)
0057            GØ TØ 9900
0058     9200   CØNTINUE
0059            WRITE(13,920) LUD
0060      920   FØRMAT (1HØ,
               C     '*ØDISK ERRØR - A FATAL ERRØR WAS RETURNED BY'
               C     ' FFIND WHILE TRYING TØ PØSITIØN' ,/,
               C     ' THE IMAGE TAPE -LUN NO. ' ,I2,
               C     ' - TØ THE STARTING SCAN LINE NUMBER ')
0061            GØ TØ 9900
0062     9300   CØNTINUE
0063            WRITE(13,930) I, LUD, IC
0064      930   FØRMAT (1HØ,
               C     '*ØDISK ERRØR - A FATAL ERRØR WAS RETURNED BY LREED '
               C     'WHILE TRYING TØ READ SCAN LINE NUMBER ', I3, /,
               C     ' ØN IMAGE TAPE NUMBER ', I2,
               C     ' CHANNEL WAS = ', I2)
0065            GØ TØ 9900
0066     9900   CØNTINUE
0067            STØP 4
0068            END
```


SUBROUTINE
RDDISK
DSET
INITIALIZE
DISK FILE
CALL HREAD
TO READ
IMAGE
HEADER FROM
DISK
FFIND
SET UP
DATA DOT
FIELD
PREST
INITIALIZE
DATA DOT
UPDATE
PROGRAM
DO LOOP
TO READ
EVERY
TENTH
SCAN
LINE
DLSKIP
CALL LREED
TO READ
SCANLINE
RENEW
TO PROCESS
1 DATA DOT
SCAN LINE
RETURN

1.5 SUBROUTINE SRDISK

```
FØRTRAN IV-PLUS V02-04          12124153    29-AUG-77          PAGE 21
IMAUPD,FTN      /IR:BLØCKS/WR
0001            SUBRØUTINE SRDISK
                C       (LUNIMA, TFØRM, JDATE,
                C        SEGBYT,
                C       SEGJD, HEADER, BUF, BUFSZ, LLD
                C       ,SW, SEGNUM, IYR, JDAY
                C       ,DSKNUM
                C       ,ENDFIL
                C       ,MAXCHN
                C       ,MXCHAN, HDWD, JDT, F11
                C       ,XTMTFG, UNIT
                C       )
0002               IMPLICIT INTEGER (A-Z)
0003               LØGICAL*1
                C       BUF(3072), HEADER(3060),
                C       SW(2)
                C       ,SEGJD(30), SEGBYT(4)
0004              LØGICAL*4 JDATE
0005               DIMENSIØN HDWD (1530)
0006               DIMENSIØN JDT (2)
0007               CØMMØN /SKIPFL/ SKIPFL
0008               CØMMØN /HCØM/
                C       SS, SF, LS, LF, NRPDS, NDSPR,
                C       NCPR, NRPC, ACCL, NC, NS,
                C       NBIT, DØI, NCAR, SVD, RSIZ,
                C       RSKIP, HSIZ, CALP, CERR
0009               CALL RDHEAD(BUF,BUFSZ,HEADER,ENDFIL,
                C       XTMTFG, UNIT)
0010               IF (NC .LE. MAXCHN) GØ TØ 9100
0011               IF(ENDFIL .GT. 0) RETURN
0012              IF (CERR .GT. 0) GØ TØ 9000
0013               SEGNUM = HDWD(34)
0014               CALL SWAP (SEGNUM)
0015               IDAY = HEADER(78)
0016               IMØN = HEADER(79)
0017               IYR = HEADER(80)
0018               CALL JULIAN (IDAY, IMØN, JDAY, IYR)
0019               ENCØDE (2, 170, SEGJD(18)) IYR
0020             . ENCØDE (3, 175, SEGJD(20)) JDAY
0021               ENCØDE (4,200,SEGJD(14)) SEGNUM
0022      170      FØRMAT(I2)
0023      175      FØRMAT(I3)
0024               JDT(1) = IYR
0025               JDT(2) = JDAY
0026               DØ 1500 NBL = 14,22
0027               IF (SEGJD(NBL) .EQ. ' ') SEGJD(NBL) = '0'
0028     1500      CØNTINUE
0029               ENCØDE (4, 200, SEGBYT) SEGNUM
0030      200      FØRMAT (I4)
0031               RETURN
0032     9000      CØNTINUE
0033               WRITE(13,900)LUNIMA
0034      900      FØRMAT (1HØ,
                C       ' SRDISK ERRØR - AN UNRECØVERABLE ERRØR WAS ENCØUNTERED '
                C       'WHILE READING THE HEADER' ,/,
                C       '  RECØRD ØN LUN ', I2)
0035               STØP 5
```

```
0036      9100 CONTINUE
0037           WRITE(13,910) MAXCHN, NC
0038       910  FORMAT (1HO,
                C     ' THIS IMAGE FILE CONTAINS MORE THAN '
                C     ,I2, ' CHANNEL.',/,
                C     ' IT CONTAINS ', I2, ' CHANNELS.',/,
                C     ' THIS FILE WILL NOT BE PROCESSES')
0039           SKIPFL = 1
0040           CALL DUSET (LUNIMA, F11, 1)
0041           MXCHAN = 1
0042           CALL TFILE (LUNIMA, 1)
0043           RETURN
0044           END
```



ORIGINAL PAGE IS
OF POOR QUALITY

1.6 SUBROUTINE JULIAN

```
FØRTRAN IV-PLUS V02-04          12:25:00    29-AUG-77          PAGE 25
IMAUPD.FTN      /IR:BLØCKS/WR
0001            SUBRØUTINE JULIAN (IDAY, IMØN, JDAY, IYR)
0002            DIMENSIØN NDAYS (12)
0003            DATA NDAYS /31, 28, 31, 30, 31, 30,
              C     31, 31, 30, 31, 30 ,31/
0004            LEAP = 1900 + IYR
0005            LEVEN = MØD(LEAP,4)
0006            IF (LEVEN .EQ. 0) NDAYS(2) = 29
0007            JDAY = 0
0008            IM = IMØN - 1
0009            DØ 1200 I = 1, IM
0010            JDAY = NDAYS(I) + JDAY
0011      1200  CØNTINUE
0012            JDAY = JDAY + IDAY
0013            NDAYS(2) = 28
0014            RETURN
0015            END
```

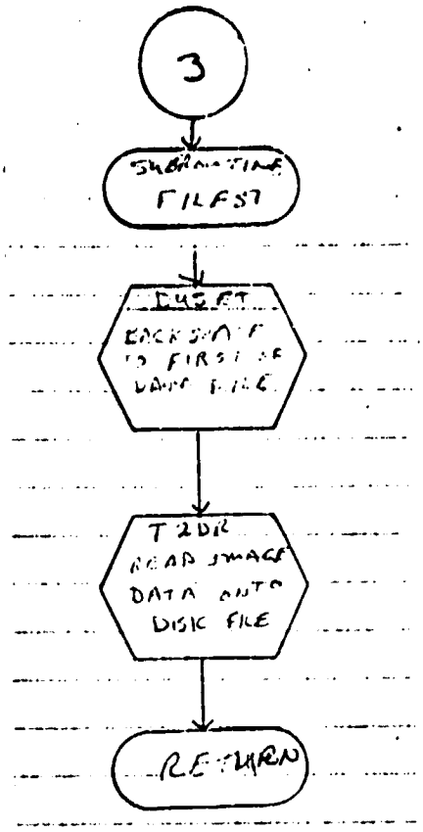
ORIGINAL PAGE IS
OF POOR QUALITY

1.6 SUBROUTINE JULIAN

A flow chart of this subroutine is not available.

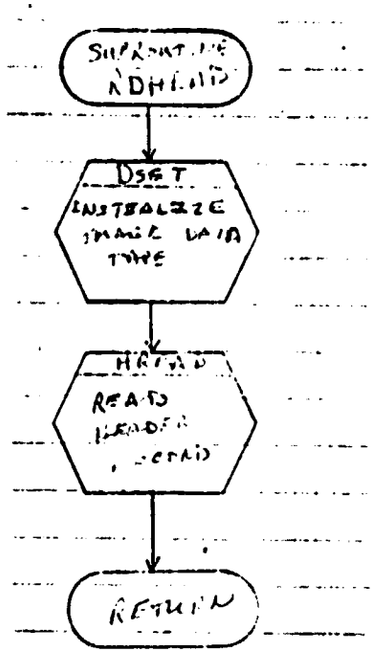
1.7 SUBROUTINE FILEST

```
FØRTRAN IV-PLUS V02-04          12:25:02    29-AUG-77          PAGE 27
IMAUPD.FTN      /TR:BLØCKS/WR
0001            SUBRØUTINE FILEST
                C     (LUNIMA, LUD, TFØRM, BUF, BUFSZ, SEGJD
                C     ,TØBDØN, NT0, MTMTFG, UNIT
                C     ,RC, PC, BC
                C     ,NØGØ
                C     ,F11, SEGNUM
                C     )
0002            IMPLICIT INTEGER (A-Z)
0003            LØGICAL*1 SEGJD(30)
0004            LØGICAL*1 BUF(BUFSZ)
0005           DIMENSIØN TØBDØN(6,20)
0006            DIMENSIØN ERR(2)
0007            DATA NSKP /0/
0008            DATA  RWD /0/
0009            RC = 0
0010            PC = 0
0011            BC = 0
0012            CALL DUSET (LUNIMA, F11, 1)
0013            CALL T2DR
                C     (LUD, SEGJD, 240, LUNIMA, MTMTFG,
                C     UNIT, NF, BUF, 3072, TFØRM, RC,
                C     PC, BC, ERR
                C     ,RWD
                C     )
0014            IF ( ERR(1) .EQ. 1) GØ TØ 9100
0015            IF (NØGØ .EQ. 1) TØBDØN(A,NT0) = BC
0016            RETURN
0017      9100  CONTINUE
0018            WRITE(13,905) ERR
0019       905  FØRMAT (1HØ, 'ERR1 =', I3,' ERR2 = ',I3)
0020            WRITE(13,910) LUD, SEGNUM
0021       910  FØRMAT (1HØ,
                C     ' FILEST ERRØR - AN UNRECØVERABLE ERRØR WAS ENCØUNTERED '
                C     'WHILE TRYING TØ READ IMAGE DATA ', /,
                C     '  FRØM THE DISK,', /,
                C     '  LUN = ' I2,/,
                C     '  SEGMENT NUMBER = ', I5)
0022            WRITE(13,915)
0023       915  FØRMAT (1HØ)
0024            STØP 6
0025            END
```


3
SUBROUTINE
FILFST
INSERT
BACKSPACE
TO FIRST OF
DATA FILE
T2DK
READ IMAGE
DATA ONTO
DISK FILE
RETURN

1.8 SUBROUTINE RDHEAD

```
FØRTRAN IV-PLUS V02-04          12125109    29-AUG-77          PAGE 29
IMAUPD.FTN      /TR:BLØCKS/WR
0001            SUBRØUTINE RDHEAD(BUF,BUFSZ,HEADER,ENDFIL,
              C     MTMTFG, UNIT)
0002            IMPLICIT INTEGER ( A-Z)
0003            LØGICAL*1 BUF, HEADER(3060)
              C     ,FILNAM
0004          DIMENSIØN BUF (3072)
0005            CØMMØN /STATUS/ S(2)
0006          CØMMØN /HC0M/
              C     SS, SE, LS, LE, NRPDS, NDSPR,
              C     NCPR, NRPC, ANGL, NC, NS,
              C     NBIT, DØI, NCAR, SVD, RSIZ,
              C     PSKIP, HSIZ, CALP, CERR
0007            CØMMØN /SKIPFL/ SKIPFL
0008          DATA LUNIMA /1/
0009            DATA SKIPFL /0/
0010            DATA RWD /0/
0011            TFØRM=1
0012            SWFG=1
0013            FN=0
0014            F11=1
0015          CALL DSET
              C     (LUNIMA, F11, FILNAM,
              C     MTMTFG, UNIT, SKIPFL, RWD)
0016            SKIPFL = 0
0017            CALL HREAD
              C     (LUNIMA,HEADER, BUF, 3072,
              C     TFØRM, F11, PRTY, IEØF, SWFG)
0018            SKIPFL = 0
0019            IF(IEØF .GT. 0) GØ TØ 2000
0020            ENDFIL=1
0021            RETURN
0022     2000   CØNTINUE
0023            RWD = 2
0024            IF(CERR .GT. 0) GØ TØ 9000
0025            RETURN
0026     9000   CØNTINUE
0027            WRITE(13,900) LUNIMA
0028      900   FØRMAT (1HØ,
              C     'RDUNIV ERRØR = A FATAL ERRØR WAS RETURNED BY HREAD '
              C     'WHILE TRYING TØ READ',/,
              C     '  THE HEADER RECØRD ØN THE IMAGE TAPE, LUN = '
              C     ,I2)
0029            WRITE(13,920)CERR
0030      920   FØRMAT(1HØ,'CERR=',I3)
0031          ST0P
0032            END
```


DSET
RETURN

1.9 SUBROUTINE PRESET

```
HF0RTRAN IV-PLUS V02-04          12:25:38   29-AUG-77          PAGE 1
DDD0T.FTN        /TR:BL0CKS/WR
0001             SUBR0UTINE PRESET(DIRWD,DIPL0G,DIRI4)
0002             IMPLICIT INTEGER(A-Z)
0003             INCLUDE 'SYI:300.31CAMSPARAM.INC'
0004 *            PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     *           1,MAXV=11,ND0TS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *           2N0SPWD=6,N0DTWD=10
0005             L0GICAL*1 LINE(NPIX,MAXCHN),VECT0R(MAXCHN),L0GREC(52),
                 *  DIRL0G(512)
0006             DIMENSI0N DDREC(26),WDL0C(6),BYL0C(6),DIRWD(256)
0007             INTEGER*4 DIRI4(128),DATES(6)
0008             EQUIVALENCE (DDREC,L0GREC)
0009             DATA WDL0C/5,9,13,17,21,25/    !W0RD L0CATI0N 0F GIB
0010             DATA BYL0C/5,13,21,29,37,45/   !BYTE L0CATI0N 0F ACQUISITI0NS
0011             DATA N0TSET/-9/
0012             SEGNUM=DIRWD(1)
0013             DF=DIRL0G(3)
0014             ACQNM=DIRL0G(4)
0015             IMDATE=DIRI4(14)
0016             D0 5 I=1,6
0017     5       DATES(I)=DIRI4(I+1)
0018             YC0RD=0
0019             A=ACQNM
0020             IF(A .EQ. 2 .0R. DF .EQ. 0) A=A+1
0021             REC=0
0022             RETURN
         C
         C
         C
0023             ENTRY RENEW(LINE)
0024             YC0RD=YC0RD+DLSKIP
0025             D0 150 I=DSSKIP,NPIX,DSSKIP
0026             REC=REC+1
0027             D0 30 J=1,MAXCHN
0028             VECT0R(J)=LINE(I,J)
0029     30      C0NTINUE
0030             IF(ACQNM) 160,40,50
0031     40      XC0RD=I
0032             L0GREC(1)=XC0RD
0033             L0GREC(2)=YC0RD
0034             D0 45 J=3,52
0035             L0GREC(J)=N0TSET
0036     45      C0NTINUE
0037             G0 T0 60
0038     50      READ(3'REC) DDREC
0039             IF(DF .GT. 0) CALL SQUEEZ(DDREC,ACQNM,DF,WDL0C)
0040     60      K=BYL0C(A)
0041             D0 65 J=1,MAXCHN
0042             L0GREC(K)=VECT0R(J)
0043             K=K+1
0044     65      C0NTINUE
0045             CALL KAUTH(VECT0R,G,B)
0046             K=WDL0C(A)
0047             DDREC(K)=G
0048             DDREC(K+1)=B
0049             WRITE(3'REC) DDREC
0050             X=IFLD(0,8,DDREC(1))
```

```
FØRTRAN IV-PLUS V02-04          12125138    29-AUG-77          PAGE 2
DDDØT,FTN       /TRIBLØCKS/WR
0051            PRINT 120,X,LØGREC(2),LØGREC(3),LØGREC(4)
0052    120     FØRMAT(1HØ,'X=',I3,2X,'Y=',I3,2X,'TYPE=',I2,2X,'LABEL=',I2)
0053            PRINT 125,(LØGREC(J),J=5,8),DDREC(5),DDREC(6),
               *(LØGREC(J),J=13,16),DDREC(9),DDREC(10),
               *  (LØGREC(J),J=21,24),DDREC(13),DDREC(14),(LØGREC(J),J=29,32),
               *  DDREC(17),DDREC(18),(LØGREC(J),J=37,40),DDREC(21),DDREC(22),
               *  (LØGREC(J),J=45,48),DDREC(25),DDREC(26)
0054    125     FØRMAT(5X,18I6)
0055    150     CØNTINUE
0056    160     RETURN
0057            END
```

ORIGINAL PAGE IS
OF POOR QUALITY

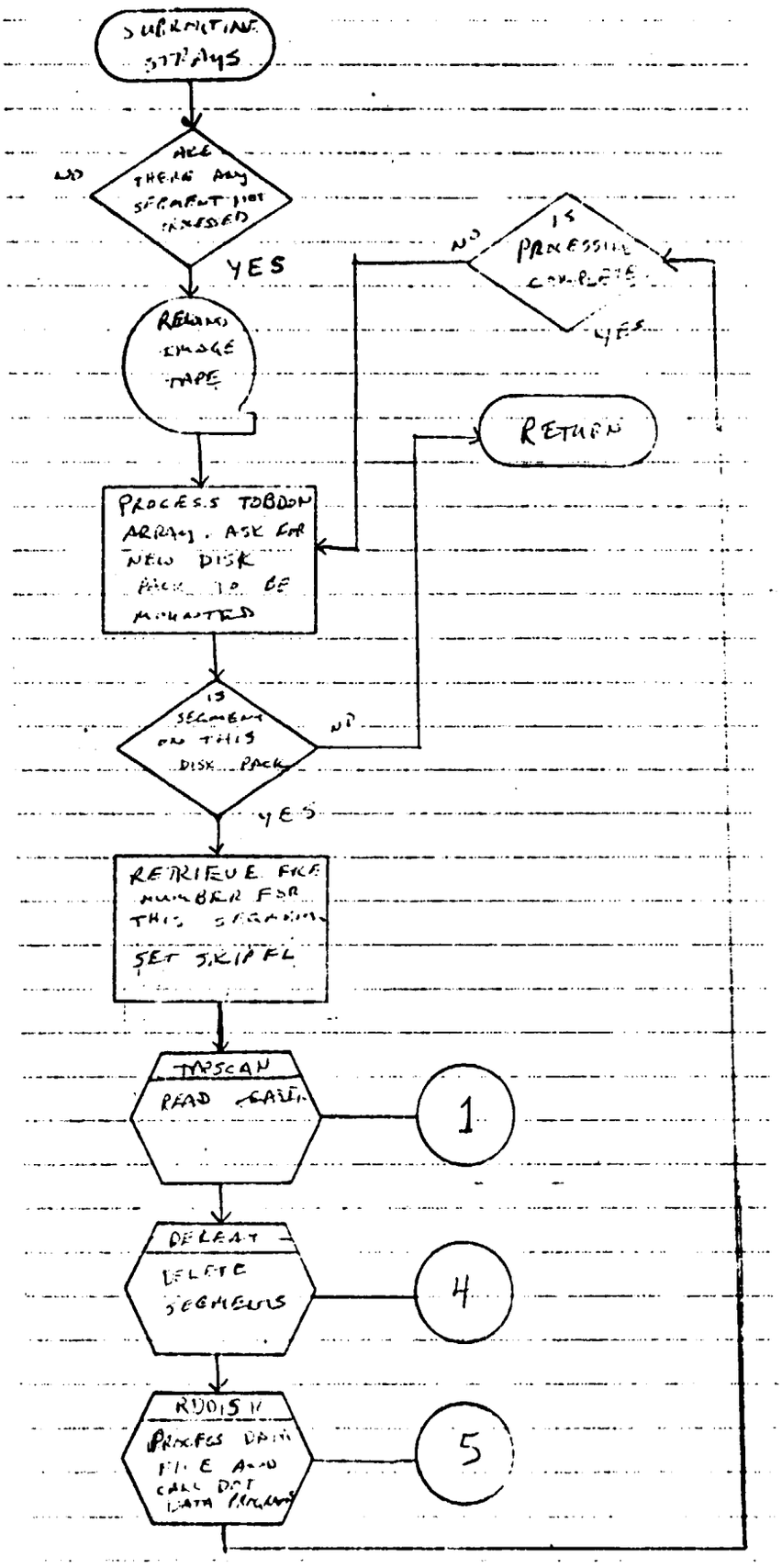
1.9 SUBROUTINE PRESET

A flow chart of this subroutine is not available.

1.10 SUBROUTINE STRAYS

```
      SUBROUTINE STRAYS
     C    (DISKON, TOBDON, DIRWRD, DIRBYT, NTO
     C     ,LUD, TFORM, HEADER,
     C     DSKNUM,
     C     BUF, BUFSZ, INDEXF, IDXBYT, SEGJD
     C     ,KDX, DLSKIP, MAXCHN, MAXACD
     C     ,LUNIMA, KLINE
     C     )
        IMPLICIT INTEGER (A-Z)
        LOGICAL*1 DIRBYT(512)
        LOGICAL*1 HEADER(3060)
        DIMENSION DISKON(10)
         DIMENSION HEAD(23)
        DIMENSION TOBDON(6,20)
        DIMENSION DIRWRD(256)
         COMMON /NEXREC/ NREC
        IF(NTO .EQ. 0) GO TO 9000
        DO 7000 KD = 1, 10
         CALL TRWD (LUNIMA)
        IF (DISKON(KD) .EQ. 0) GO TO 7000
        NTRY = 0
 6000   CONTINUE
         PRINT 600, DISKON(KD)
  600   FORMAT ('  MOUNT DISK PACK NUMBER ', I4)
         PAUSE 'MOUNT NEW DISK'
         READ (4, 605) HEAD
  605   FORMAT (23A2)
         DECODE (4,610, HEAD) DSKNUM
  610   FORMAT (I4)
        IF (DSKNUM .EQ. DISKON(KD))  GO TO 6300
        PRINT 620, DSKNUM
  620   FORMAT (1HO,
     C     '  DISK PACK ', I5, ' HAS BEEN MOUNTED')
        IF (NTRY .EQ. 0) GO TO 6250
        PRINT 625, DISKON(KD)
  625   FORMAT (1HO,
     C     '  DISK PACK HEADER HAS NOT CONTAINED THE CORRECT '
     C    'DISK PACK NUMBER =', I5,
     C     ' - AFTER TWO TRIES',/,
     C     '  PROGRAM WILL CONTINUE')
        GO TO 7000
 6250   CONTINUE
        NTRY = 1
        GO TO 6000
 6300   CONTINUE
        DO 6600 KF = 1, NTO
        IF (TOBDON(1,KF) .NE. KD) GO TO 6600
         CALL TAPSCN (SEGNUM, TOBDON(2,KF))
        CALL RELEAT
     C     (SEGNUM, DFLAG, JDATE , ACONUM, DIRBYT,
     C     DIRLG4, DIRWRD, DRLINE, HEADER
     C     ,MAXACD
     C     )
         ENCODE (4, 630, SG) SEGNUM
  630   FORMAT (I4)
        CALL RDDISK
```

```
C      (LUD, TFØRM, JDATE, SEGJD,
C      HEADER, BUF, BUFSZ, INDEXF, INXBYT,
C      DIRWRD, DIRBYT, DIRLG4
C      , KDX, DLSKIP, MAXCHN
C      ,MAXACD, NDØTS, SG
C      ,SEGNUM
C      )
       WRITE (2'DRLINE) DIRWRD
        WRITE (3'1) INDEXF
        AQ = ACQNUM
        DFG = DFLAG
        WRITE (11) SEGNUM, IYR, JDAY, DYR, DDAY,
C         NRECR, NPRTYR, NBLØCK, AQ, DFG,
C         GREEN, ANG, AZM
        KLINE = KLINE + 1
6600   CØNTINUE
7000   CØNTINUE
9000   CØNTINUE
       RETURN
       END
```


SUBROUTINE SYSTEMS
ARE THERE ANY SEGMENTS NEEDED
NO
YES
READ IMAGE TAPE
PROCESS TOBDON ARRAY. ASK FOR NEW DISK PACK TO BE MOUNTED
IS SEGMENT ON THIS DISK PACK
NO
YES
RETRIEVE FILE NUMBER FOR THIS SEGMENT
SET SKIP FL
TMSCAN
READ TAPE
1
DELEAT
DELETE SEGMENTS
4
RUDISK
PROCESS DATA FILE AND CALL DAT
5
IS PROCESSING COMPLETE
NO
YES
RETURN

1.11 SUBROUTINE TAPSCN

```
      SUBROUTINE TAPSCN (SEGNUM, TBDSEG)
      IMPLICIT INTEGER (A-Z)
      COMMON /SKIPFL/ SKIPFL
2000  CONTINUE
      CALL SRDISK (LUNIMA, TFORM, JDATE, SEGBYT,
     C    SEGJD, HEADER, BUF, BUFSZ, LUD, SW,
     C    SEGNUM, IYR, JDAY, DSKNUM, ENDFIL, MAXCHN,
     C    MXCHAN, HDWD, JDT, F11)
      IF (SEGNUM .EQ. TBDSEG) GO TO 3000
      CALL DUSET(LUNIMA, F11, 1)
      SKIPFL = 1
3000  CONTINUE
      RETURN
      END
```

1.11 SUBROUTINE TAPSCN

A flow chart of this subroutine is not available.

1.12 SUBROUTINE SQUEEZ

```
FØRTRAN IV-PLUS V02-04          12:25:48    29-AUG-77          PAGE 5
DDDØT,FTN       /TR:BLØCKS/WR
0001            SUBRØUTINE SQUEEZ(DDREC,ACQNM,DF,LØC)
0002            IMPLICIT INTEGER(A-Z)
0003            DIMENSIØN DDREC(96),LØC(6)
0004            WDLØC1=LØC(DF)
0005            WDLØC2=LØC(DF+1)
0006            NBYTES=(ACQNM-DF)*6     !NUMBER ØF BYTES TØ LEFT ADJUST
      C                                 WHEN THE DELETED ACQUISITIØN IS REMØVED
0007            IF(NBYTES .LE. 0)RETURN
0008            CALL SUBSTR(DDREC(WDLØC2),1,NBYTES,DDREC(WDLØC1),1,NBYTES)
0009            RETURN
0010            END
```

1.13 SUBROUTINE IFLD

```
FORTRAN IV-PLUS V02-04          12:25:50     29-AUG-77          PAGE 7
DD:0T.FTN                /TR:FL:CKS/WR
0001                      FUNCTION IFLD(BITP:S,NBITS,NWRD)
0002                     INTEGER BITP:S,WORD
0003                     DIMENSION MASK(16)
0004                     DATA MASK/1,3,7,15,31,63,127,255,511,1023,2047,4095,8191,16383,
                    1 32767,"177777/
0005                     NWD=ISHFT(WORD,-BITP:S)
0006                     IFLD=IAND(NWD,MASK(NBITS))
0007                     RETURN
0008                     END
```

1.13 SUBROUTINE IFLD

A flow chart of this subroutine is not available.

1.14 SUBROUTINE KAUTH

```
FORTRAN IV-PLUS V02-04          12:25:51    29-AUG-77          PAGE 9
DDDGT.FT'       /TR:BLOCKS/WR
0001            SUBROUTINE KAUTH(V,G,S)
0002            INTEGER G,P
0003            LOGICAL*1 V(4)
0004            DIMENSION X(4),H(4,4),FI(4),IY(4)
0005            DATA H/.438,.632,.586,.264,
     *                 -.290,-.562,.400,.491,
     *                 -.824,-.573,-.050,.185,
     *                  .223,.019,-.543,.809/
      C
0006            DATA FI/0.0,16.0,25.0,20.0/
      C
0007            DO 20 I=1,4
0008      20    X(I)=V(I)
      C
0009            DO 40 I=1,2
0010            G=0.0
0011            DO 30 K=1,4
0012            G=G + H(K,I) * V(K)
0013      30    CONTINUE
0014            IY(I)=G + FI(I) + .5
0015      40    CONTINUE
      C
0016            G=IY(1)
0017            S=IY(2)
0018            RETURN
0019            END
```

KAUTH
SI GDVC

1.14 SUBROUTINE KAUTH

A flow chart of this subroutine is not available.

1.15 SUBROUTINE T2DR

```
      SUBROUTINE T2DR(FLUN,FN,NBLCK,TLUN,MXF,UNIT,NF,BUF,BUFSZ,FORM,RC,
     1 PC,BC,ERR, RWD)
      IMPLICIT INTEGER (A-Z)
      BYTE FN(33)
      COMMON /STATUS/ S(2)
        COMMON /IOLP/IO,LP
      INTEGER BUF(1),ERR(2)
        INTEGER SS(2)
        LSTLNG = 0
      DO 1 I = 1,33
      IF(FN(I),EQ,' ') GO TO 2
1     IF(FN(I),EQ,0) GO TO 2
2     CALL FVOPEN(2,FLUN,FN,I-1,S,FLUN+20,NBLCK)
      IF(S(1),NE,0) GO TO 5
      CALL TINIT(TLUN,MXF,UNIT)
      CALL TATCH(TLUN)
      CALL TFILE(TLUN,NF)
3       PRCT=20
33      CALL TREAD(TLUN,BUF(7),BUFSZ)
      IF(S(1),EQ,"365) GO TO 4
      IF(S(1),EQ,"374) GO TO 7
      IF(S(1),NE,1) GO TO 8
      RC = RC + 1
      GO TO 9
7       PRCT = PRCT - 1
        IF(PRCT,LE,0) GO TO 77
        CALL TSKIP(LUN,-1)
        GO TO 33
77      PC = PC + 1
      RC = RC + 1
9     CALL FVDLST(FLUN,BUF(5),BUF(6))
      CALL FVNXRC(FLUN,S(2),BUF(1),BUF(2))
      BUF(3)=FORM
      BUF(4)=S(2)
        LSTLNG=S(2)
      CALL FVWRIT(FLUN,BUF,S(2)+12)
      CALL FVWAIT(FLUN,S)
      IF(S(1),NE,1) GO TO 5
      GO TO 7
5     ERR(1) = 1
      ERR(2) = S(1)
        GO TO 10
8     ERR(1) = 2
      ERR(2) = S(1)
        GOTO 10
4     CALL FVDLST(FLUN,SS(1),SS(2))
        CALL FVDGET(FLUN,ERR(1),ERR(2))
        BC = BC + ERR(1) - 1
D       WRITE(IO,1000)SS(1),SS(2)
      CALL FVDSET(FLUN,SS(1),SS(2))
D       WRITE(IO,1000)SS(1),SS(2)
      Q = IMIN0(LSTLNG,512)
D       WRITE(IO,1000)Q
      CALL FVREAD(FLUN,BUF,Q)
        CALL FVWAIT(FLUN,ERR)
D       WRITE(IO,1000)Q
D       WRITE(IO,1000)(BUF(I),I=1,512)
      BUF(1)= 0
      BUF(2) = 0
D       WRITE(IO,1000)(BUF(I),I=1,512)
      CALL FVDSET(FLUN,SS(1),SS(2))
D       WRITE(IO,1000)SS(1),SS(2)
      CALL FVWRIT(FLUN,BUF,Q)
        CALL FVWAIT(FLUN,ERR)
```

```
D       WRITE(10,1000)Q
C       WRITE(10,1000)(BUF(I),I=1,512)
D1000   FORMAT((' ',8(B6,1X)))
        ERR(1)=0
        ERR(2)=0
   10   CONTINUE
        IF (RWD .EQ. 0) GO TO 9000
        CALL TRWD(TLUN)
 9000  CONTINUE
        CALL TRLSE(TLUN)
        CALL FVCLOS(FLUN)
        RETURN
        END
```

1.15 SUBROUTINE T2DR

A flow chart of this subroutine is not available.

2. DO/DU FIELD UPDATE PROGRAM MDODU

```
      PROGRAM MD0DU
CCC
C
C        D0/DU FILE UPDATE
C        READS AND PROCESSES DEL FOSTER CARDS
C
CCC
      COMMON /COM1/ IACTV,IACDI,NEWL,INDEXP(4),IERFLG,KCKBP,
     1 IDLTE,NDEXR,ID0DU
      BYTE ID0DU
      COMMON /COM2/ ID,IDFLD(6),ICARD(80),ITYPE
      BYTE IDFLD,ICARD,ITYPE
      PARAMETER NBDR=512,NBTBL=400,NBD0U=1450
        PARAMETER NWDR=256
      PARAMETER NBD0UE=30
      COMMON /BUFRS/ IXDIR,IBDIR,IXINX,IXD0V,
     1  LDTBL(NBTBL),LDIR(NBDR),LD0DU(NBD0U)
      BYTE LDIR,LD0DU
      COMMON /COM3/ NFLD,NC00RD
      COMMON /FILES/ IN,I0UT,LU1,LU2,LU3,LU4,M0UNT
C        LTYPE - CARD TYPES TO BE PROCESSED
        INCLUDE 'SY:[300,3]CAMSPARAM.INC'
        BYTE ISTRT
      BYTE  LTYPE(20), INXTY(7)
      BYTE I0NE
      BYTE LDATE(9),LTIME(8)
      DATA LTYPE /
     1  1HS,1HE,1HG,1HD,1HE,1HL,
     2 1HS,1HE,1HG,
     3 1HD,1HE,1HL,1HD,1HU,1H+,
     4 1H/,1H/,1HE,1HN,1HD/
      DATA INXTY /6,3,3,1,1,1,5/
        T1=SECNDS(0.0)
        CALL TIME (LTIME)
        CALL DATE (LDATE)
 7000 FORMAT (21X,'C A M S  I - 1 0 0   D A T A  B A S E ',
     1 'T R A N S A C T I O N  R E P O R T',//,
     2  81X,'DATE '9A1,/,81X,'TIME  ',8A1,//,
     3  35X,'FIELD DEFINITION DATA BASE UPDATE ',
     4  'PROGRAM / MAY 77',//)
      IBDIR=67
C        INITIALIZE NO. OF FIELDS
      NFLD=0
      IN=4
      I0UT=10
C        LU2 - DIRECTORY FILE
      LU2=8
        DEFINE FILE LU2 (200,NWDR,U,NEXTR)
C        LU1 - D0/DU FILE
      LU1=9
C        LU3 - DOT DATA FILE
        LU3=11
        DEFINE FILE LU3 (ND0TS,26,U,NEXTD)
C        INITIALIZE ACTIVITY FLAG FOR D0/DU FILE
      IACTV=0
C        INITIALIZE ACTIVITY FLAG FOR DIRECTORY AND INDEX FILES
      IACDI=0
C        INITIALIZE TO EXPECT SEGMENT START OR DELETE
      INDEXP(1)=1
      DO 90 I=2,4
      INDEXP(I)=0
   90 CONTINUE
        CALL ASSIGN (4,'CARD.DAT')
      CALL ASSIGN (I0UT,'0UTFILE.DAT')
```

ORIGINAL PAGE IS OF POOR QUALITY

```
        WRITE (IOUT,7000) LDATE,LTIME
        CALL ASSIGN (LU2,'DB0:[300,300]DIRFILE.LAT')
   94 CONTINUE
  100 CONTINUE
C        BEGINNING OF CARD READ AND PROCESS LOOP
      READ (4,1000,ERR=990,END=990) ICARD
 1000 FORMAT (80A1)
        WRITE (IOUT,1001) (ICARD(I),I=1,80)
 1001 FORMAT (1X,80A1)
      NEWL=0
C        IS THIS CARD TYPE TO BE PROCESSED
      ISTRT=1
      IONE=1
      DO 104 I=1,7
        IRTN=INDORT(LTYPE,ISTRT,INXTY(I),ICARD,IONE,INXTY(I))
        IF (IRTN .GT. 0) GO TO 110
      ISTRT=ISTRT+INXTY(I)
  104 CONTINUE
      IF (NEWL .NE. 0) WRITE (IOUT,5000)
 5000 FORMAT (1H )
        WRITE (IOUT,1104)
      NEWL=1
 1104 FORMAT (1H+,80X,'UNKNOWN CARD TYPE')
C        CARD READ IS NOT TO BE PROCESSED
      GO TO 100
  110 CONTINUE
      ITYPE=1
      NEWL=0
      GO TO (200,201,300,300,300,400,500), ITYPE
  200 CONTINUE
C        CONTROL CARD (START OF SEGMENT OR DELETE)
C        CLEAR CHECK AND BYPASS FLAG
      KCHBP=0
      ITYPE=2
      GO TO 202
  201 CONTINUE
      ITYPE=1
  202 CONTINUE
C        WAS CONTROL CARD (START OF SEGMENT OR DELETE) EXPECTED
      IF (INDEXP(1) .EQ. 0) GO TO 280
C        YES
      CALL CNTRL
      GO TO 100
  280 CONTINUE
C        CONTROL CARD WAS NOT EXPECTED
      CALL NCNTRL
      GO TO 100
  300 CONTINUE
C        WAS FIELD START CARD EXPECTED
      IF (INDEXP(2) .EQ. 0) GO TO 350
      CALL FLDST
      GO TO 100
  350 CONTINUE
      CALL NFLDST
      GO TO 100
  400 CONTINUE
C        WAS FIELD (COORDINATE) CARD EXPECTED
      IF (INDEXP(3) .EQ. 0) GO TO 450
      CALL FIELD
      GO TO 100
  450 CONTINUE
      CALL NFIELD
      GO TO 100
  500 CONTINUE
C        WAS END OF SEGMENT CARD EXPECTED
      IF (INDEXP(4) .EQ. 0) GO TO 550
```

```
        IF (INDEXP(3) ,NE, 0) CALL FLDEND
      CALL SEGEND
      GØ TØ 100
  550 CØNTINUE
      CALL NSEGND
      GØ TØ 100
  900 CØNTINUE
C        END ØF FILE READ = WRAPUP
        IF (INDEXP(4) ,EQ, 0) GØ TØ 901
        IF (INDEXP(3) ,EQ, 2) CALL FLDEND
        CALL NSEGND
        GØ TØ 902
  901   CØNTINUE
        IF (INDEXP(1) ,NE, 0) GØ TØ 902
      CALL SEGEND
  902   CØNTINUE
        ELAPS=SECNDS(T1)
        T2=ELAPS
        JSCR=T2/60,
        SCR=JSCR*60,
        ISEC=T2-SCR+0,5
        IHR=JSCR/60
        IMIN=JSCR-IHR*60
        WRITE (IØUT,1222) IHR,IMIN,ISEC
 1222   FØRMAT (1HØ,4X,'TØTAL ELAPSED TIME= ',I2,':',I2,':',I2)
        TYPE 1900
      WRITE (IØUT,1900)
 1900 FØRMAT (1HØ,'NØRMAL TERMINATIØN')
      CALL EXIT
  990 CØNTINUE
      WRITE (IØUT,1990)
 1990 FØRMAT (81X,'ERRØR READING CARD')
      GØ TØ 100
      END
```

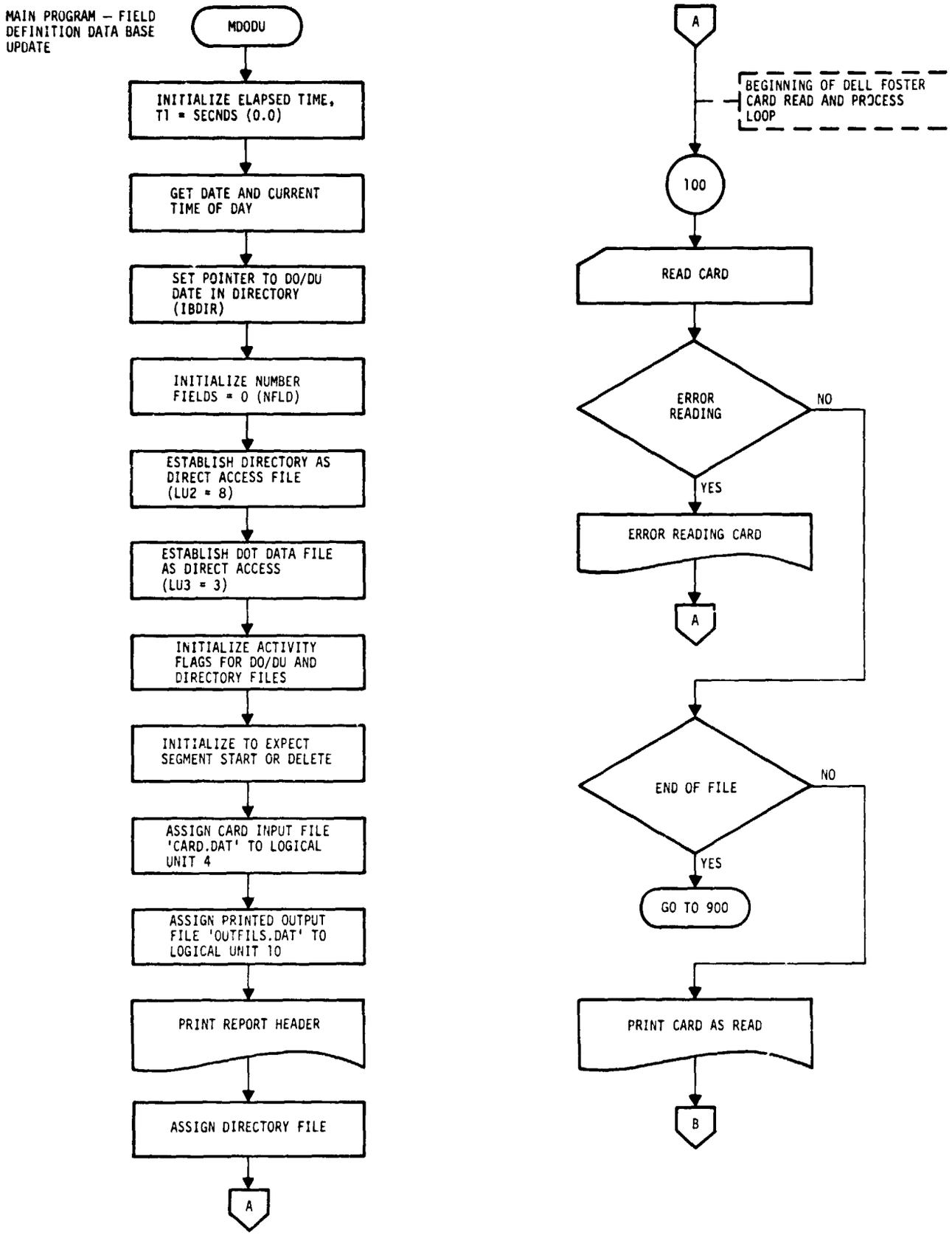
2.1 MAIN PROGRAM (FLDUPD)

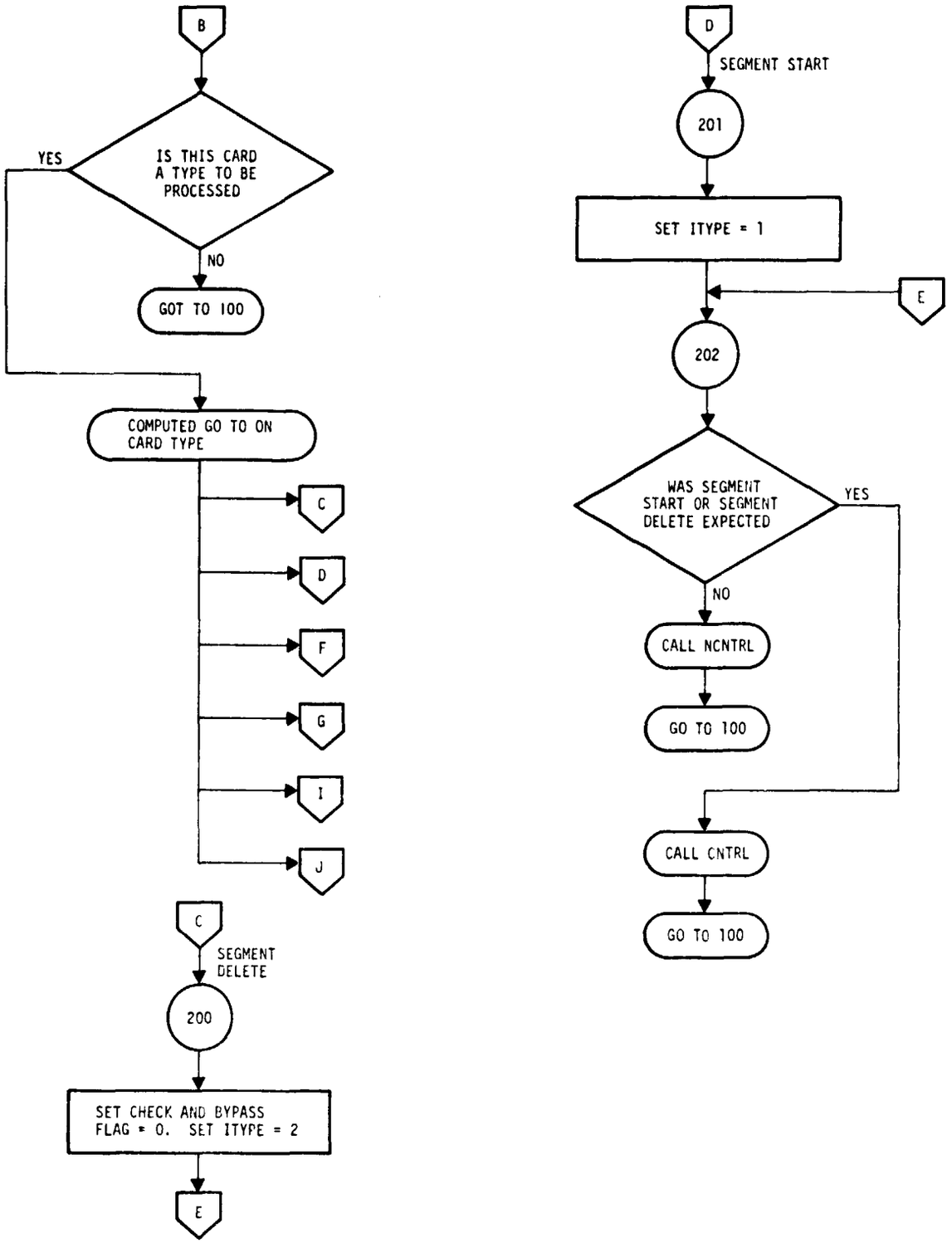
```
HF0RTRAN IV-PLUS V02-04                    08:46:33    11-MAY-77          PAGE 1
FLDUPD.FTN      /TR:BL0CKS/WR
0001           PR0GRAM MD0DU
               CCC
               C
               C         D0/DU FILE UPDATE
               C         READS AND PR0CESSES DEL F0STER CARDS
               C
               CCC
0002            C0MM0N /C0M1/ IACTV,IACDI,IEWL,INDEXP(4),IERFLG,KCKBP,
                1 IDLTE,NDEXR,ID0DU
0003            BYTE ID0DU
0004            C0MM0N /C0M2/ ID,IDFLD(6),ICARD(80),ITYPE
0005            BYTE IDFLD,ICARD,ITYPE
0006            PARAMETER NBDR=512,NBTBL=400,NBD0L=1450
0007              PARAMETER NWDR=256
0008            PARAMETER NBD0UE=30
0009            C0MM0N /BUFRS/ IXDIR,IBDIR,IXINX,IXD0U,
                1  LDTBL(NBTBL),LDIR(NBDR),LD0DU(NBD0U)
0010            BYTE LDIR,LD0DU
0011            C0MM0N /C0M3/ NFLD,NC00RD
0012            C0MM0N /FILES/ IN,I0UT,LU1,LU2,LU3,LU4,M0UNT
               C         LTYPE - CARD TYPES T0 BE PR0CESSED
0013              INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0014 *          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     *          1,MAXV=11,ND0TS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *          2N0SPWD=6,N0DTWD=10
0015              BYTE ISTRT
0016            BYTE  LTYPE(20), INXTY(7)
0017            BYTE IBNE
0018            BYTE LDATE(9),LTIME(8)
0019            DATA LTYPE /
                1  1HS,1HE,1HG,1HD,1HE,1HL,
                2 1HS,1HE,1HG,
                3 1HD,1HE,1HL,1HD,1HU,1H*,
                4 1H/,1H/,1HE,1HN,1HD/
0020            DATA INXTY /6,3,3,1,1,1,5/
0021              T1=SECNDS(0.0)
0022              CALL TIME (LTIME)
0023              CALL DATE (LDATE)
0024       7000 F0RMAT (21X,'C A M S  I - 1 0 0   D A T A  B A S E ',
                1 'T R A N S A C T I O N  R E P 0 R T',//,
                2  81X,'DATE '9A1,/,81X,'TIME  ',8A1,//,
                3  35X,'FIELD DEFINITI0N DATA BASE UPDATE ',
                4  'PR0GRAM / MAY 77',//)
0025            IBDIR=67
               C         INITIALIZE N0. 0F FIELDS
0026            NFLD=0
0027            IN=4
0028            I0UT=7
               C         LU2 - DIRECT0RY FILE
0029            LU2=8
0030              DEFINE FILE LU2 (200,NWDR,U,NEXTR)
               C         LU1 - D0/DU FILE
0031            LU1=9
               C         LU3 - D0T DATA FILE
0032              LU3=3
0033              DEFINE FILE LU3 (ND0TS,26,U,NEXTD)
```

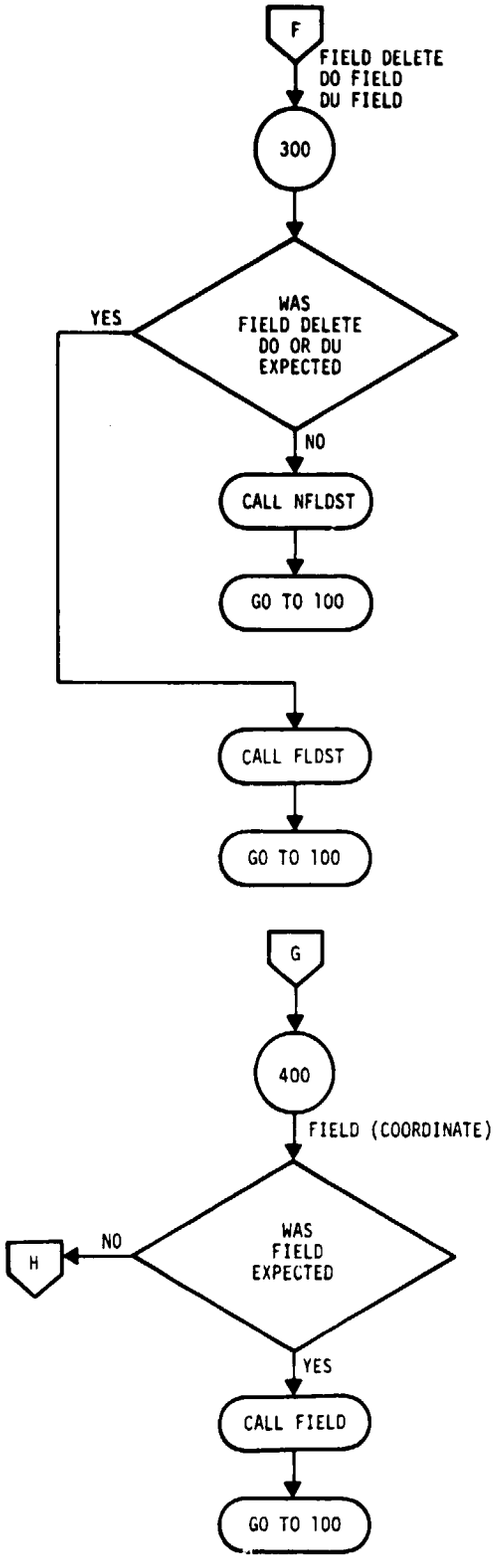
```
FLDUPD,FTN      /TRIBLØCKS/WR
      C          INITIALIZE ACTIVITY FLAG FØR D0/DU FILE
0034            IACTV=0
      C          INITIALIZE ACTIVITY FLAG FØR DIRECTØRY AND INDEX FILES
0035            IACDI=0
      C          INITIALIZE TØ EXPECT SEGMENT START ØR DELETE
0036            INDEXP(1)=1
0037            DØ 90 I=2,4
0038            INDEXP(I)=0
0039         90 CØNTINUE
0040              CALL ASSIGN (4,'CARD.DAT')
0041            CALL ASSIGN (7,'ØUTFILE.DAT')
0042              WRITE (7,7000) LDATE,LTIME
0043              CALL ASSIGN (LU2,'DBDI[300,300]DIRFILE.DAT')
0044         94 CØNTINUE
0045        100 CØNTINUE
      C          BEGINNING ØF CARD READ AND PRØCESS LØØP
0046            READ (4,1000,ERR=990,END=900) ICARD
0047       1000 FØRMAT (80A1)
0048              WRITE (7,1001) (ICARD(I),I=1,80)
0049       1001 FØRMAT (1X,80A1)
0050            NEWL=0
      C          IS THIS CARD TYPE TØ BE PRØCESSED
0051            ISTRT=1
0052            IØNE=1
0053            DØ 104 I=1,7
0054              IRTN=INDART(LTYPE,ISTRT,INXTY(I),ICARD,IØNE,INXTY(I))
0055              IF (IRTN .GT. 0) GØ TØ 110
0056            ISTRT=ISTRT+INXTY(I)
0057        104 CØNTINUE
0058            IF (NEWL .NE. 0) WRITE (7,5000)
0059       5000 FØRMAT (1H )
0060              WRITE (7,1104)
0061            NEWL=1
0062       1104 FØRMAT (1H+,80X,'UNKNØWN CARD TYPE')
      C          CARD READ IS NØT TØ BE PRØCESSED
0063            GØ TØ 100
0064        110 CØNTINUE
0065            ITYPE=I
0066            NEWL=0
0067            GØ TØ (200,201,300,300,300,400,500), ITYPE
0068        200 CØNTINUE
      C          CØNTRØL CARD (START ØF SEGMENT ØR DELETE)
      C          CLEAR CHECK AND BYPASS FLAG
0069            KCKBP=0
0070            ITYPE=2
0071            GØ TØ 202
0072        201 CØNTINUE
0073            ITYPE=1
0074        202 CØNTINUE
      C          WAS CØNTRØL CARD (START ØF SEGMENT ØR DELETE) EXPECTED
0075            IF (INDEXP(1) .EQ. 0) GØ TØ 280
      C          YES
0076            CALL CNTRL
0077            GØ TØ 100
0078        280 CØNTINUE
      C          CØNTRØL CARD WAS NØT EXPECTED
```

```
FLDUPD.FTN      /TRIBLØCKS/WR
0079            CALL NCNTRL
0080            GØ TØ 100
0081          300 CØNTINUE
              C         WAS FIELD START CARD EXPECTED
0082            IF (INDEXP(2) .EQ. 0) GØ TØ 350
0083            CALL FLDST
0084            GØ TØ 100
0085          350 CØNTINUE
0086            CALL NFLDST
0087            GØ TØ 100
0088          400 CØNTINUE
              C         WAS FIELD (CØØRDINATE) CARD EXPECTED
0089            IF (INDEXP(3) .EQ. 0) GØ TØ 450
0090            CALL FIELD
0091            GØ TØ 100
0092          450 CØNTINUE
0093            CALL NFIELD
0094            GØ TØ 100
0095          500 CØNTINUE
              C         WAS END ØF SEGMENT CARD EXPECTED
0096            IF (INDEXP(4) .EQ. 0) GØ TØ 550
0097              IF (INDEXP(3) .NE. 0) CALL FLDEND
0098            CALL SEGEND
0099            GØ TØ 100
0100          550 CØNTINUE
0101            CALL NSEGND
0102            GØ TØ 100
0103          900 CØNTINUE
              C         END ØF FILE READ - WRAPUP
0104              IF (INDEXP(4) .EQ. 0) GØ TØ 901
0105              IF (INDEXP(3) .EQ. 2) CALL FLDEND
0106              CALL NSEGND
0107              GØ TØ 902
0108           901    CØNTINUE
0109              IF (INDEXP(1) .NE. 0) GØ TØ 902
0110            CALL SEGEND
0111           902    CØNTINUE
0112              ELAPS=SECNDS(T1)
0113              WRITE (7,1899) ELAPS
0114          1899 FØRMAT (1H0,'ELAPSED TIME ',1PE12.6)
0115              TYPE 1900
0116            WRITE (7,1900)
0117          1900 FØRMAT (1H0,'NØRMAL TERMINATIØN')
0118            CALL EXIT
0119           990 CØNTINUE
0120            WRITE (7,1990)
0121          1990 FØRMAT (A1X,'ERRØR READING CARD')
0122            GØ TØ 100
0123            END
```

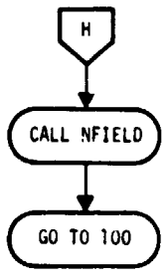
2.1 MAIN PROGRAM — FLDUP

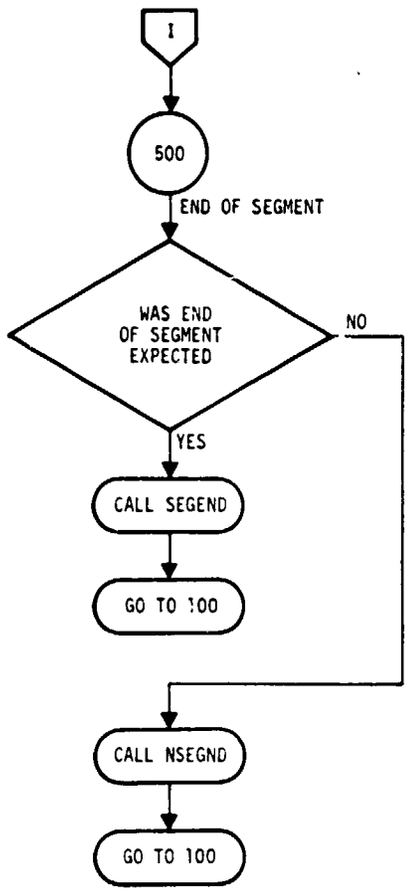


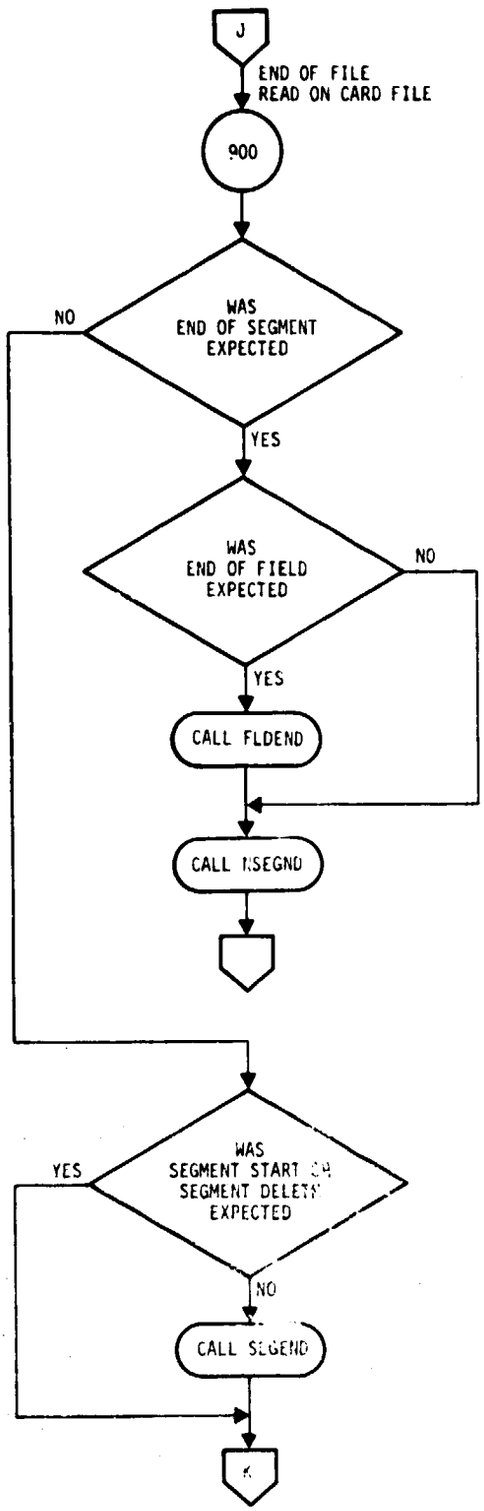

B
IS THIS CARD A TYPE TO BE PROCESSED
YES
NO
GOT TO 100
COMPUTED GO TO ON CARD TYPE
C
D
F
G
I
J
C
SEGMENT DELETE
200
SET CHECK AND BYPASS FLAG = 0. SET ITYPE = 2
E
D
SEGMENT START
201
SET ITYPE = 1
E
202
WAS SEGMENT START OR SEGMENT DELETE EXPECTED
YES
NO
CALL NCNTRL
GO TO 100
CALL CNTRL
GO TO 100

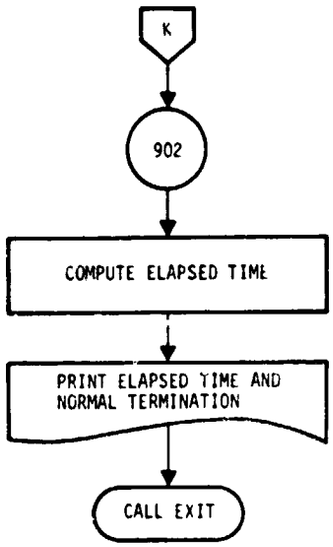

F
FIELD DELETE
DO FIELD
DU FIELD
300
WAS
FIELD DELETE
DO OR DU
EXPECTED
YES
NO
CALL NFLDST
GO TO 100
CALL FLDST
GO TO 100

G
400
FIELD (COORDINATE)
WAS
FIELD
EXPECTED
NO
H
YES
CALL FIELD
GO TO 100


H
CALL NFIELD
GO TO 100


I
500
END OF SEGMENT
WAS END
OF SEGMENT
EXPECTED
NO
YES
CALL SEGEND
GO TO 100
CALL NSEGND
GO TO 100


J
END OF FILE
READ ON CARD FILE
900
WAS
END OF SEGMENT
EXPECTED
NO
YES
WAS
END OF FIELD
EXPECTED
NO
YES
CALL FLDEND
CALL NSEGND
WAS
SEGMENT START OR
SEGMENT DELETE
EXPECTED
YES
NO
CALL SEGEND
K


K
902
COMPUTE ELAPSED TIME
PRINT ELAPSED TIME AND
NORMAL TERMINATION
CALL EXIT

2.2 SUBROUTINE CNTRL

```
FORTRAN IV+PLUS V02-04          C8146153    11-MAY-77          PAGE 6
FLDUPD.FTN      /TR:BLOCKS/HR
0001            SUBROUTINE CNTRL
                CCC
                C
                C       CONTROL CARD (SEGSTART OR DELETE) READ
                C
                CCC
0002                COMMON /COM1/ IACTV,IACDI,NEWL,INDEXP(4),IERFLG,KCKBP,
                   1 IDLTE,NDEXR,ID0DU
0003                BYTE ID0DU
0004                COMMON /COM2/ ID,IDFLD(6),ICARD(80),ITYPE
0005                BYTE IDFLD,ICARD,ITYPE
0006                COMMON /COM3/ NFLD,NC00RD
0007                PARAMETER NBDR=512,NBTBL=400,NBD0U=1450
0008                PARAMETER NBD0UE=30
0009                COMMON /BUFRS/ IXDIR,IRDIR,IXINX,IXD0U,
                   1  LDTBL(NBTBL),LDIR(NBDR),LD0DU(NBD0U)
0010                BYTE LDIR,LD0DU
0011                COMMON /FILES/ IN,I0UT,LU1,LU2,LU3,LU4,M0UNT
0012                COMMON /COMNAM/ NAMFIL(28),NAMFLD(9),NAMD0T(9),IUIC(9)
0013                  DIMENSION I2NAM(14)
0014                  EQUIVALENCE (NAMFIL,I2NAM)
0015                  INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0016 *              PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     *             1,MAXV=11,ND0TS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *             2N0SPWD=6,N0DTWD=10
0017                  PARAMETER NBD0T=10868,NBPD0T=52
0018                  COMMON /COMD0T/ IXD0T,LD0T(NBD0T)
0019                  BYTE LD0T
0020                BYTE IDA(4)
0021                BYTE IBLNK,IC0MMA
0022                BYTE IZER0
0023                BYTE NAMFIL,NAMFLD,NAMD0T,IUIC
0024                BYTE ID0,ID9
0025                  BYTE LB,LK,L1,L2
0026                  DATA LB/1HB/, LK/1HK/ ,L1/1H1/,  L2/1H2/
0027                DATA IZER0 /0/
0028                DATA IBLNK /1H /, IC0MMA /1H,/
                C         NAMFLD = COMPLETE D0/DU NAME
0029                DATA NAMFLD /1HF,1HI,1HE,1HL,1HD,
                   1  1H ,1HD,1HA,1HT/
                C         NAMD0T = COMPLETE D0T DATA FILE NAME
0030                DATA NAMD0T /1HD,1HD,1H0,1HT,1HS,1H ,
                   1  1HD,1HA,1HT/
                C         IUIC   = UIC, USER IDENTIFICATION CODE
0031                DATA IUIC /1H[,1H3,1H0,1H0,1H,,1H3,1H0,1H0,1H]/
0032                DATA NAMFIL /1HD,1HB,1H2,1H:,1H ,1H ,1H ,1H ,
                   1  1H ,1H ,1H ,1H ,1H ,1H ,1H ,1H ,1H ,
                   2  1H ,1H ,1H ,1H ,1H ,1H ,1H ,1H ,
                   3  1H ,1H ,1H /
0033                DATA ID0 /'0'/, ID9 /'9'/
                C       INITIALIZE CHECK AND BYPASS
0034                  KCKBP=0
0035                  IERFLG=0
                C         SET DELETE FLAG
0036                IDLTE=0
0037                IF (ITYPE .EQ. 2) IDLTE=1
```

```
0038            IB=6
0039          210 CONTINUE
0040            GO TO (211,214), ITYPE
0041          211 CONTINUE
0042            IB=3
              C     SET SWITCH TO EXPECT FIELD START (FLDSTART)
0043            DO 212 I=1,4
0044            INDEXP(I)=0
0045          212 CONTINUE
0046            INDEXP(2)=1
0047          214 CONTINUE
              C     GET ID
              C
0048            IB=IB+1
0049            IF (IB .GT. 72) GO TO 100
0050            IF (ICARD(IB) .EQ. IBLNK) GO TO 214
0051            IF (ICARD(IB) .EQ. ICOMMA) GO TO 214
0052            J=IB-1
0053            DO 215 I=1,4
0054            IDA(I)=ICARD(I+J)
0055            IF (IDA(I) .LT. ID0 .OR. IDA(I) .GT. ID9)
               1  GO TO 900
0056          215 CONTINUE
0057              NAMFIL(14)=IDA(1)
0058              NAMFIL(15)=IDA(2)
0059              NAMFIL(16)=IDA(3)
0060              NAMFIL(17)=IDA(4)
0061            DECODE (4,1215,IDA,ERR=900) II
0062          1215 FORMAT (I4)
0063            WRITE (7,5000)
0064            NEWL=1
              C     SEGMENT ID VALIDITY CHECK
0065            IF (ICARD(J+5) .EQ. IBLNK .OR.
               1  ICARD(J+5) .EQ. ICOMMA) GO TO 220
              C     INVALID
0066            IF (NEWL .NE. 0) WRITE (7,5000)
0067          5000 FORMAT (1H )
0068            WRITE (7,1214)
0069          1214 FORMAT (1H+,20X,'SEGMENT ID OUT OF RANGE')
0070            NEWL=1
0071          216 CONTINUE
0072            IF (ITYPE .EQ. 2) GO TO 260
0073            MOKBP=1
0074            GO TO 100
0075          220 CONTINUE
0076          222 CONTINUE
0077              IFLG=0
0078              CALL DSKCHK(ID,IXDIR,IDDSK,IFLG)
0079              IF (IFLG .EQ. 0) GO TO 230
0080              IF (IFLG .EQ. 2) GO TO 224
0081              IF (NEWL .NE. 0) WRITE (7,5000)
0082              WRITE (7,1223)
0083          1223 FORMAT (1H+,20X,'SEGMENT NOT ON CURRENT DISK PACK')
0084              NEWL=1
0085              GO TO 216
0086          224 CONTINUE
```

```
FLDUPD,FTN      /TR;BL0CKS/WR
0087            IF (NEWL .NE. 0) WRITE (7,5000)
0088          WRITE (7,1224)
0089     1224 FØRMAT (1H+,80X,'SEGMENT NØT IN DIRECTØRY TABLE')
0090          NEWL=1
0091          GØ TØ 210
0092      230 CØNTINUE
0093            IF (IDDSK .LT. 1000) GØ TØ 232
0094            NAMFIL(2)=LK
0095            NAMFIL(3)=L1
0096            GØ TØ 234
0097      232 CØNTINUE
0098            NAMFIL(2)=LB
0099            NAMFIL(3)=L2
          C       IXDIR - RECØRD NØ. ØF DATA IN DIRECTØRY FILE
0100      234 CØNTINUE
          C       READ DIRECTØRY
0101            CALL RDDIR (ID,IERR)
0102            IF (IERR .EQ. 0) GØ TØ 250
0103            KCKBP=1
0104            GØ TØ 100
0105      250 CØNTINUE
          C       EXISTENCE FLAGS FRØM DIRECTØRY
0106          IXDØT=0
0107          IXIST=0
0108          IF (LDIR(IBDIR) .NE. I2ERØ) IXIST=1
0109          IF (LDIR(IBDIR+4) .NE. I2ERØ) IXDØT=1
          C       DØES DØT DATA FILE EXIST FØR SEGMENT
0110          IF (IXDØT .NE. 0) GØ TØ 245
0111          IF (NEWL .NE. 0) WRITE (7,5000)
0112          WRITE (7,1248) ID
0113     1248 FØRMAT (1H+,80X,'NØ DØT DATA FILE FØR SEGMENT ',I4)
0114          NEWL=1
0115          GØ TØ 248
0116      245 CØNTINUE
          C       STØRE DØT FILE NAME
0117          J=18
0118          K=5
0119          DØ 2450 I=1,9
0120          NAMFIL(J)=NAMDØT(I)
0121          J=J+1
          C       STØRE UIC
0122          NAMFIL(K)=IUIC(I)
0123          K=K+1
0124     2450 CØNTINUE
0125            NAMFIL(27)=I2ERØ
0126            CALL ASSIGN (LU3,I2NAM)
          C       READ DØT DATA FILE
0127            CALL RDDØT (ID,IERR)
0128      248 CØNTINUE
          C       DØES DØ/DU FILE EXIST FØR SEGMENT
0129          IF (IXIST .NE. 0) GØ TØ 252
0130          IF (ITYPE .NE. 2) GØ TØ 252
0131          IF (NEWL .NE. 0) WRITE (7,5000)
0132          WRITE (7,1252) ID
0133     1252 FØRMAT (1H+,80X,'NØ DØ/DU FILE FØR SEGMENT ',I4)
0134          NEWL=1
```

ORIGINAL PAGE
OF POOR QUALITY

```
0135          GØ 10 260
0136       252 CØNTINUE
0137          J=18
0138          K=5
0139          DØ 2520 I=1,9
0140          NAMFIL(J)=NAMFLD(I)
0141          NAMFIL(K)=IUIC(I)
0142          J=J+1
0143          K=K+1
0144       2520 CØNTINUE
0145            NAMFIL(27)=IZERØ
0146            IF (IXIST .NE. 0) GØ TØ 2522
          C      ASSIGN DØ/DU ILE SEGMENT
0147            ØPEN (UNIT=LU1,NAME=I2NAM,TYPE='NEW',
               1  FØRM='UNFØRMATTED',ERR=2521)
0148            GØ TØ 2522
0149       2521 CØNTINUE
0150            WRITE (7,2300) (NAMFIL(I),I=1,27)
0151       2300 FØRMAT (81X,'ERRØR-FILE ØPEN',/,81X,27A1)
0152            KCKBP=1
0153            GØ TØ 100
0154       2522 CØNTINUE
0155            IF (IXIST .EQ. 0) GØ TØ 100
0156            ØPEN (UNIT=LU1,NAME=I2NAM,TYPE='ØLD',
               1  FØRM='UNFØRMATTED',ERR=2521)
0157            IF (ITYPE .EQ. 2) GØ TØ 254
          C      READ DØ/DU FILE
0158          CALL RDDØDU (ID,IERR)
0159            IF (IERR .NE. 0) KCKBP=1
0160          GØ TØ 100
0161        254 CØNTINUE
          C       DELETE DØ/DU FILE
0162            CLØSE (UNIT=LU1,DISPØSE='DELETE',ERR=2540)
0163            IACTV=0
0164            GØ TØ 2541
0165       2540 CØNTINUE
0166            WRITE (7,1260) (NAMFIL(I),I=1,27)
0167       1260 FØRMAT (81X,'ERRØR-FILE DELETE',/,81X,27A1)
0168       2541 CØNTINUE
0169            N=IHDIR-1
0170          DØ 256 I=1,4
0171            LDIR(N+I)=IZERØ
0172        256 CØNTINUE
0173          IACDI=1
          C       CHECK FØR DØ/DU IN DØT DATA FILE AND SET TØ Ø
0174            CALL UPDØT(1)
0175            CALL SEGEND
0176        260 CØNTINUE
0177          IB=IB+4
0178          IF (IB .LE. 72) GØ TØ 214
0179        100 CØNTINUE
0180          RETURN
0181        900 CØNTINUE
0182          IF (NEWL .NE. 0) WRITE (7,5000)
0183          NEWL=1
          C       SET CHECK AND BYPASS FLAG
```

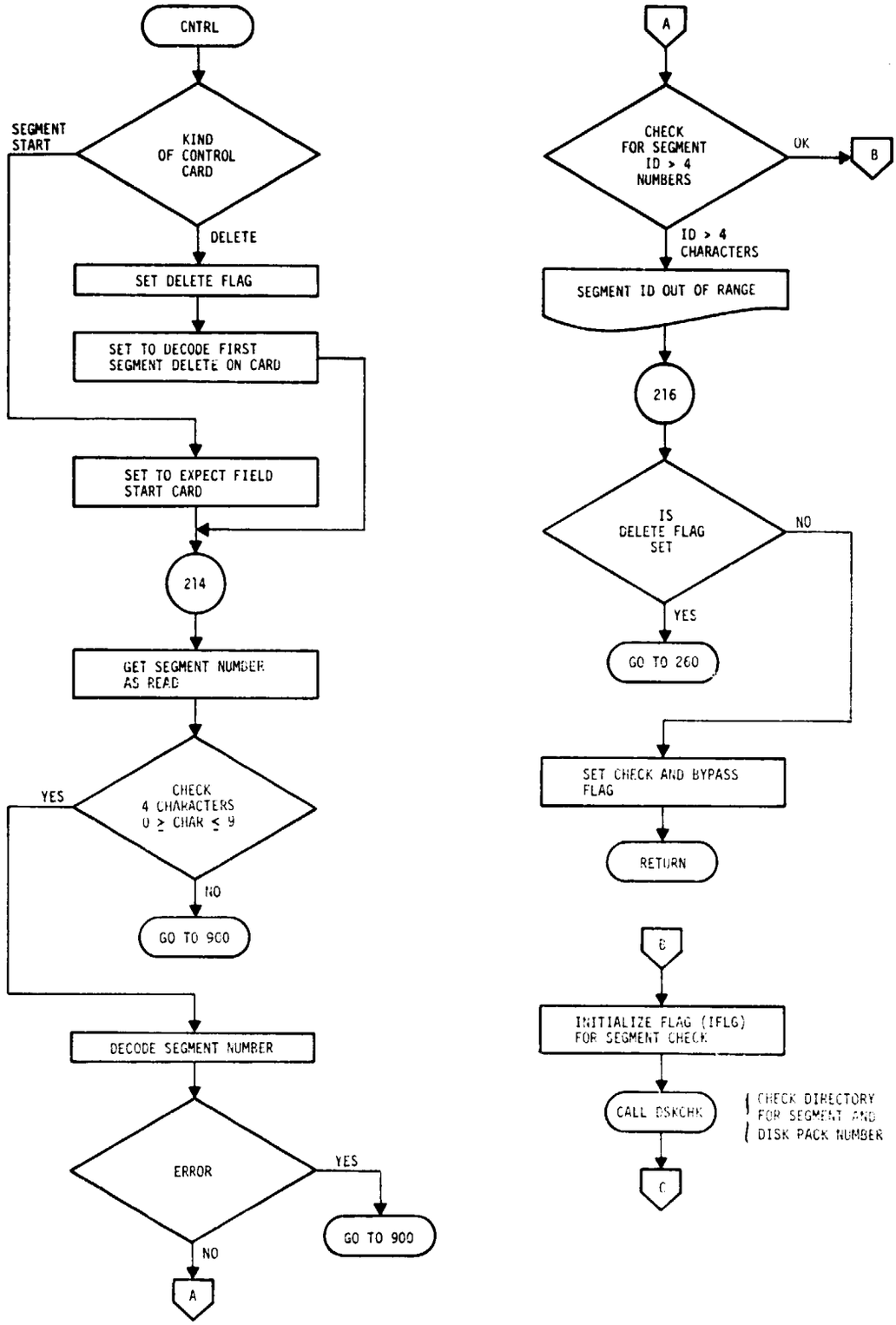
```
FLDUPD,FTN      /TRIBLØCKS/WR
0184            KCKBP=1
0185            IF (IDLTE .EQ. 0) GØ TØ 100
0186            GØ TØ 260
0187          902 CØNTINUE
              C     ERRØR TRYING TØ ØPEN DØT DATA FILE
0188              WRITE (7,1902)
0189         1902 FØRMAT (81X,'ERRØR-DØT DATA FILE ØPEN')
0190              KCKBP=1
0191              GØ TØ 248
0192            END
```

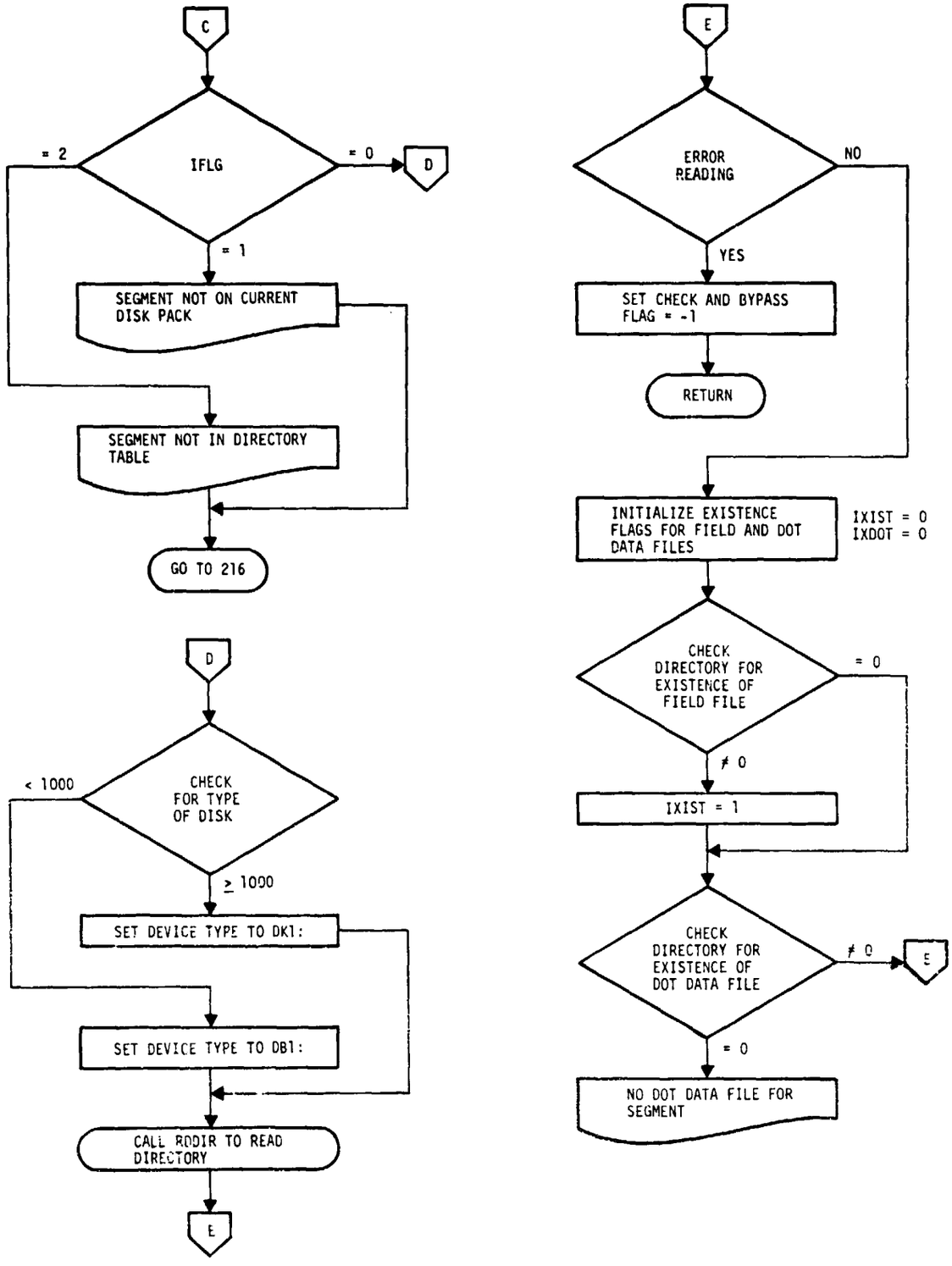
ORIGINAL PAGE IS
OF POOR QUALITY

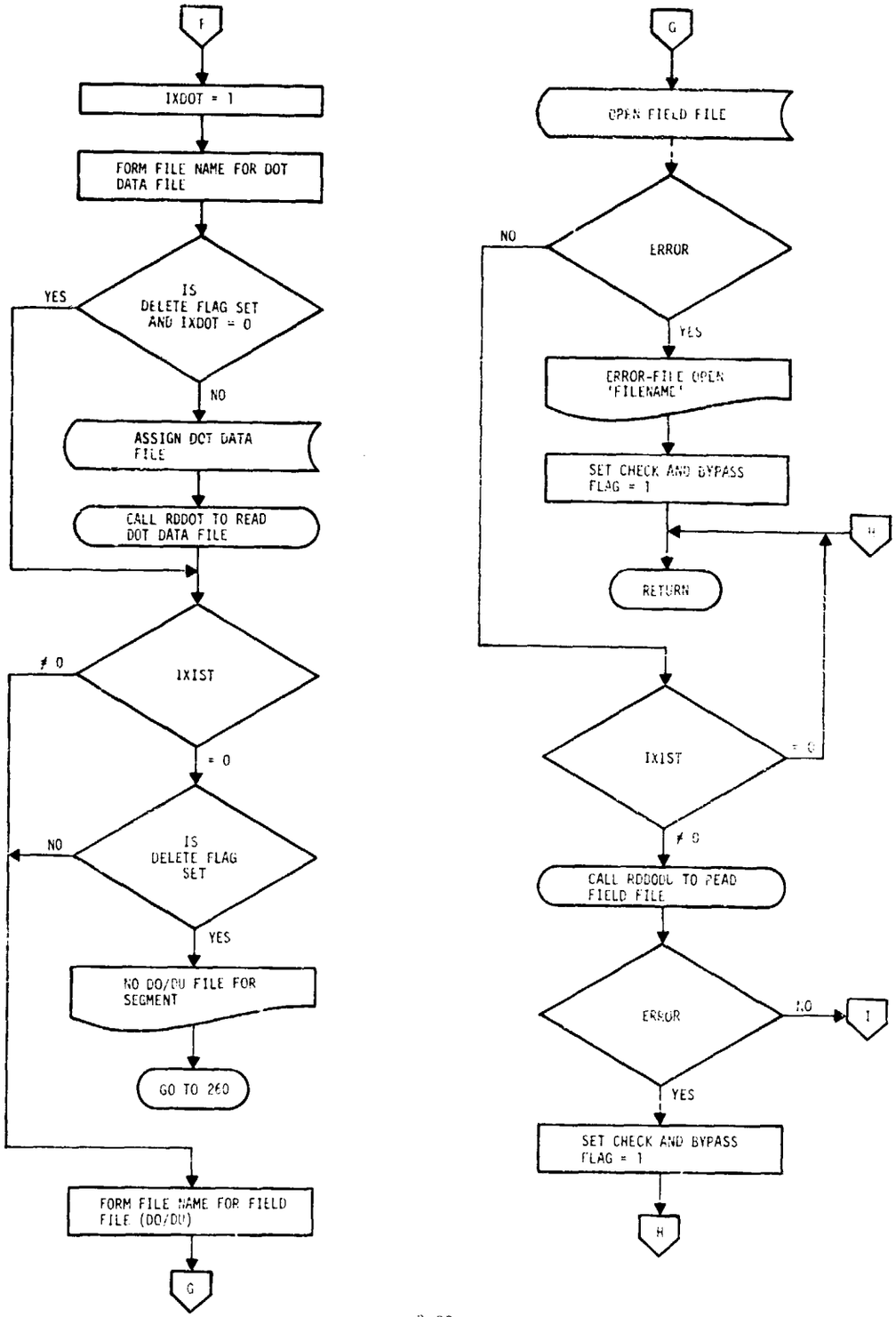
```
FLDUPD.FTN      /TR:BL@CKS/WR
0001            FUNCTI@N INDBRT(A,I,N,B,J,M)
      C  FIND B IN A
0002            BYTE I,N,J,M
0003            BYTE A,B
0004            INDBRT=0
0005            K=N+1-M
0006            IF(K.LT.I)RETURN
0007            D@ 1 L=I,K
0008            IF(K3MBRT(A,L,N,B,J).NE.0)G@ T@ 1
0009            INDBRT=L
0010            RETURN
0011          1 C@NTINUE
0012            RETURN
0013            END
```

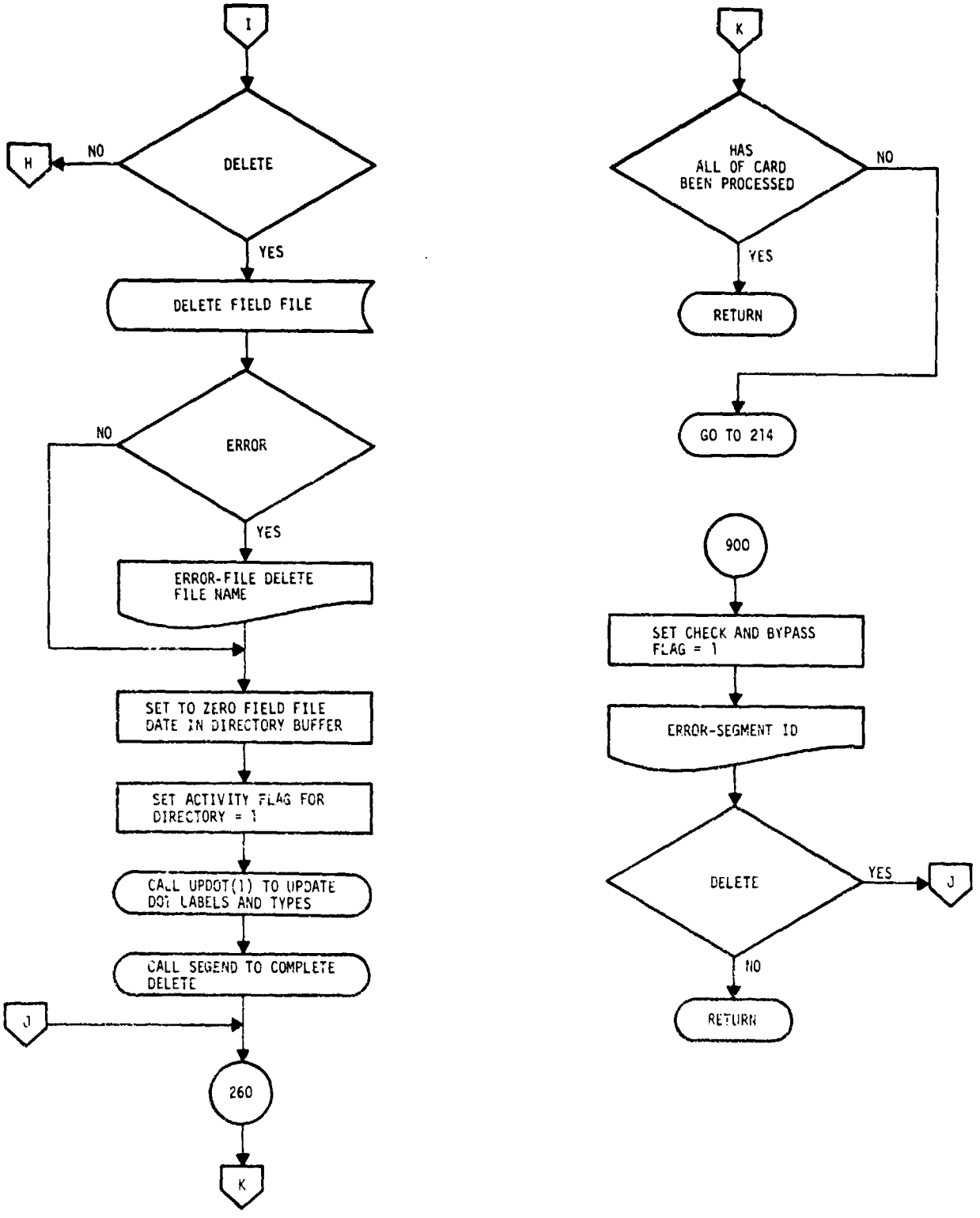
```
FLDUPD.FTN      /TR:BLØCKS/WR
0001            FUNCTIØN KØMBRT(A,I,N,B,J)
C  A.LT.B  -1
C  A.EQ.B   0
C  A.GT.B  +1
0002            BYTE I,N,J
0003            LØGICAL*1 A(1),B(1)
0004            IS=I
0005            JS=J
0006            KØMBRT=0
0007            DØ 1 K=1,N
0008            M1=A(IS)
0009            M2=B(JS)
0010            IS=IS+1
0011            JS=JS+1
0012            IF(M1-M2)2,1,3
0013          2 KØMBRT=-1
0014            RETURN
0015          3 KØMBRT=+1
0016            RETURN
0017          1 CØNTINUE
0018            RETURN
0019            END
```

2.2 SUBROUTINE CNTRL




C
IFLG
= 2
= 0
D
= 1
SEGMENT NOT ON CURRENT DISK PACK
SEGMENT NOT IN DIRECTORY TABLE
GO TO 216
D
CHECK FOR TYPE OF DISK
< 1000
≥ 1000
SET DEVICE TYPE TO DK1:
SET DEVICE TYPE TO DB1:
CALL RDDIR TO READ DIRECTORY
E
E
ERROR READING
NO
YES
SET CHECK AND BYPASS FLAG = -1
RETURN
INITIALIZE EXISTENCE FLAGS FOR FIELD AND DOT DATA FILES
IXIST = 0
IXDOT = 0
CHECK DIRECTORY FOR EXISTENCE OF FIELD FILE
= 0
≠ 0
IXIST = 1
CHECK DIRECTORY FOR EXISTENCE OF DOT DATA FILE
≠ 0
E
= 0
NO DOT DATA FILE FOR SEGMENT

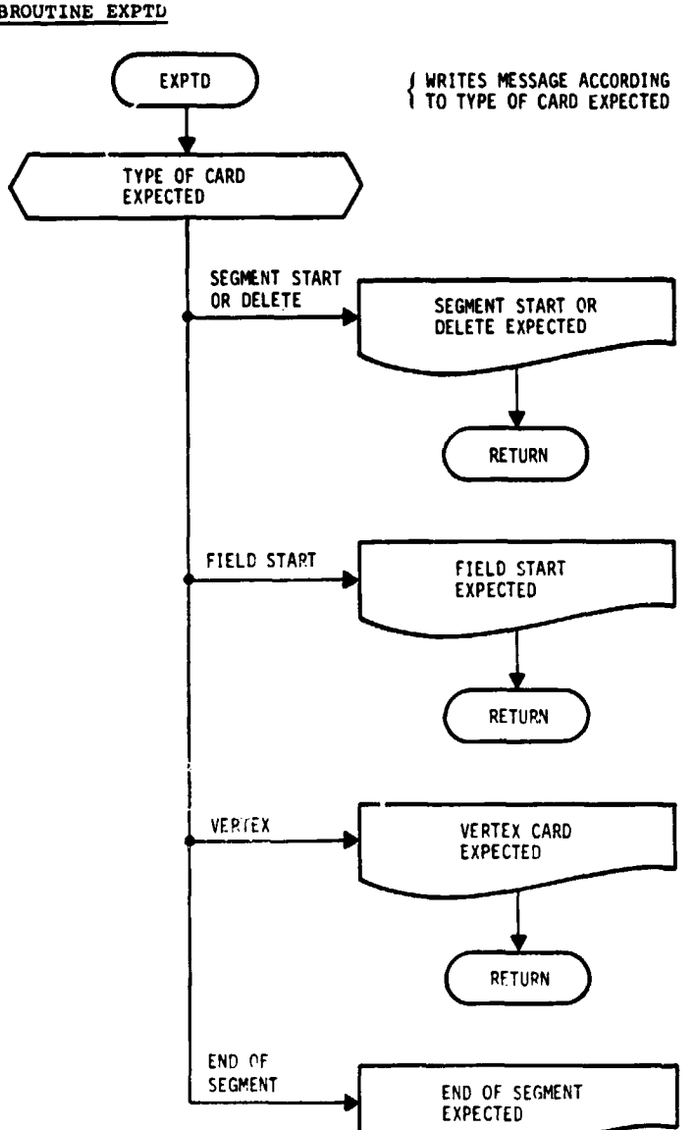

F
IXDOT = 1
FORM FILE NAME FOR DOT DATA FILE
IS DELETE FLAG SET AND IXDOT = 0
YES
NO
ASSIGN DOT DATA FILE
CALL RDDOT TO READ DOT DATA FILE
IXIST
≠ 0
= 0
IS DELETE FLAG SET
NO
YES
NO DO/DU FILE FOR SEGMENT
GO TO 260
FORM FILE NAME FOR FIELD FILE (DO/DU)
G
G
OPEN FIELD FILE
ERROR
NO
YES
ERROR-FILE OPEN 'FILENAME'
SET CHECK AND BYPASS FLAG = 1
RETURN
IXIST
= 0
≠ 0
CALL RDDODU TO READ FIELD FILE
ERROR
NO
YES
SET CHECK AND BYPASS FLAG = 1
H


I
DELETE
NO
H
YES
DELETE FIELD FILE
ERROR
NO
YES
ERROR-FILE DELETE
FILE NAME
SET TO ZERO FIELD FILE
DATE IN DIRECTORY BUFFER
SET ACTIVITY FLAG FOR
DIRECTORY = 1
CALL UPDOT(1) TO UPDATE
DOT LABELS AND TYPES
CALL SEGEND TO COMPLETE
DELETE
J
260
K
K
HAS
ALL OF CARD
BEEN PROCESSED
NO
YES
RETURN
GO TO 214
900
SET CHECK AND BYPASS
FLAG = 1
ERROR-SEGMENT ID
DELETE
YES
J
NO
RETURN

2.3 SUBROUTINE EXPTD

```
FØRTRAN IV-PLUS V02-04          08:47:29    11-MAY-77          PAGE 17
FLDUPD.FTN      /TRIBLØCKS/WR
0001            SUBRØUTINE EXPTD(I)
                CCC
                C
                C         WRITES MESSAGE ACCØRDING TØ TYPE ØF CARD EXPECTED
                C
                CCC
0002                COMMØN /CØM1/ IACTV,IACDI,NEWL,INDEXP(4),IERFLG,KCKBP,
                   1 IDLTE,NDEXR,IDØDU
0003                 BYTE IDØDU
0004                COMMØN /CØM3/ NFLD,NCØØRD
0005                COMMØN /FILES/ IN,IØUT,LU1,LU2,LU3,LU4,MØUNT
0006                IF (NEWL .NE. 0) WRITE (7,5000)
0007             5000 FØRMAT (1H )
0008                GØ TØ (10,20,30,40), I
0009               10 CØNTINUE
0010                WRITE (7,1010)
0011             1010 FØRMAT (1H+,8ØX,'SEGMENT START ØR DELETE EXPECTED,')
0012                GØ TØ 500
0013               20 CØNTINUE
0014                WRITE (7,1020)
0015             1020 FØRMAT (1H+,8ØX,'FIELD START EXPECTED')
0016                GØ TØ 500
0017               30 CØNTINUE
0018                WRITE (7,1030)
0019             1030 FØRMAT (1H+,8ØX,'VERTEX CARD EXPECTED,')
0020                GØ TØ 500
0021               40 CØNTINUE
0022                WRITE (7,1040)
0023             1040 FØRMAT (1H+,8ØX,'END ØF SEGMENT EXPECTED,')
0024              500 CØNTINUE
0025                NEWL=1
0026                RETURN
0027                END
```

2.3 SUBROUTINE EXPTD



2.4 SUBROUTINE FLDST

```
FØRTRAN IV+PLUS V02-04          08:47:33    11-MAY-77          PAGE 19
FLDUPD.FTN      /TR:BLOCKS/WR
0001            SUBRØUTINE FLDST
                CCC
                C
                C       FIELD START (FLDSTART) CARD WAS READ
                C
                CCC
                C       MAXFLD - MAXIMUM TØTAL DØ/DU FIELDS PER SEGMENT
0002            CØMMØN /CØM1/ IACTV,IACDI,NEWL,INDEXP(4),IERFLG,KCKBP,
                1 IDLTE,NDEXR,IDØDU
0003            BYTE IDØDU
0004            CØMMØN /CØM2/ ID,IDFLD(6),ICARD(80),ITYPE
0005            BYTE IDFLD,ICARD,ITYPE
0006              INCLUDE 'SYI[300,3]CAMSPARAM.INC'
0007 *          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     *          1,MAXV=11,NDØTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *          2NØSPWD=6,NØDTWD=10
0008            CØMMØN /CØM3/ NFLD,NCØØRD
0009            PARAMETER NBDR=512,NRTBL=400,NRDØL=1450
0010            PARAMETER NBDØUE=30
0011            CØMMØN /BUFRS/ IXDIR,IRDIR,IXINX,IXDØU,
                1 LDTBL(NRTBL),LDIR(NBDR),LDØDU(NBDØU)
0012            BYTE LDIR,LDØDU
0013            CØMMØN /FILES/ IN,IØUT,LU1,LU2,LU3,LU4,MØUNT
0014            CØMMØN /CØMNAM/ NAMFIL(28),NAMFLD(9),NAMDØT(9),IUIC(9)
0015              DIMENSIØN I2NAM(14)
0016              EQUIVALENCE (NAMFIL,I2NAM)
0017              BYTE NAMFIL,NAMFLD,NAMDØT,IUIC
0018            BYTE ID1,ID2,ID3,ID4
0019            BYTE IDA(4)
0020            BYTE IZERØ
0021            BYTE IBLNK,ICØMMA
0022            DATA IBLNK/1H /, ICØMMA/1H,/
0023            DATA IZERØ /0/
0024            DATA ID1 /'A'/, ID2/'Z'/,ID3/'0'/,ID4/'9'/
0025         10 CØNTINUE
0026              DØ 9 I=1,6
0027              IDFLD(I)=IBLNK
0028          9 CØNTINUE
0029              IF (ITYPE .EQ. 3) GØ TØ 11
0030            IDLTF=0
0031            J=0
0032            GØ TØ 14
0033         11 CØNTINUE
             C     SET DELETE FIELD FLAG
0034            IDLTF=1
0035            IB=3
0036         12 CØNTINUE
0037            IB=IB+1
0038            IF (IB .GT. 72) GØ TØ 500
0039            IF (ICARD(IB) .EQ. IBLNK) GØ TØ 12
0040            IF (ICARD(IB) .EQ. ICØMMA) GØ TØ 12
0041            J=IB-1
0042         14 CØNTINUE
0043              I=1
0044         15 CØNTINUE
0045              IF (ICARD(J+I) .EQ. IBLNK) GØ TØ 30
```


...NAL PAGE IS
...QUALITY

```
0046            IDFLD(I)=ICARD(J+I)
0047              I=I+1
0048              IF (I .LE. 6) GØ TØ 15
0049              IF (ICARD(J+I) .NE. IBLNK .AND.
                1  ICARD(J+I) .NE. ICØMMA) GØ TØ 35
0050           3Ø CØNTINUE
0051              NEWL=1
                C         IS FIELD ID LEGAL
                C         IS THERE MØRE THAN 1 CHARACTER
0052              IF (IDFLD(2) .EQ. IBLNK) GØ TØ 35
0053            DØ 34 I=3,6
0054              IF (IDFLD(I) .EQ. IBLNK) GØ TØ 40
0055              IF (IDFLD(I) .GE. ID1 .AND. IDFLD(I) .LE. ID2) GØ TØ 34
0056              IF (IDFLD(I) .GE. ID3 .AND. IDFLD(I) .LE. ID4) GØ TØ 34
0057              GØ TØ 35
0058           34 CØNTINUE
0059              GØ TØ 40
0060           35 CØNTINUE
0061              IF (NEWL .NE. 0) WRITE (7,5000)
0062         5000 FØRMAT (1H )
0063              WRITE (7,1035)
0064         1035 FØRMAT (1H+,80X,'ILLEGAL FIELD ID')
0065              NEWL=1
0066              IERFLG=1
0067           40 CØNTINUE
                C         IS DELETE FLAG SET
0068              IF (IDLTF .NE. 0) GØ TØ 90
0069                IF (NFLD .EQ. 0) GØ TØ 45
0070                NTØT=(NFLD+1)*NBDØUE
0071                DØ 43 J=1,NTØT,NBDØUE
0072                K=J
0073                DØ 41 I=1,6
0074                IF (IDFLD(I) .NE. LDØDU(K)) GØ TØ 43
0075                K=K+1
0076           41   CØNTINUE
                C          DUPLICATE ID'S
0077                KCKBP=1
0078                IERFLD=1
0079                WRITE (7,1410)
0080         1410 FØRMAT (81X,'DUPLICATE FIELD IDS')
0081                GØ TØ 900
0082           43 CØNTINUE
0083           45 CØNTINUE
                C         SET TØ EXPECT CØØRDINATE CARD
0084              DØ 42 I=1,4
0085              INDEXP(I)=0
0086           42 CØNTINUE
0087              INDEXP(3)=1
                C         INITIALIZE NØ. ØF VERTICE CARDS
0088              NCØØRD=0
                C         UP TØTAL NØ. ØF FIELDS
0089              IF (NFLD+1 .LE. MAXFLD) GØ TØ 48
0090              IF (NEWL .NE. 0) WRITE (7,5000)
0091              WRITE (7,1048) MAXFLD
0092         1048 FØRMAT (1H+,80X,'NØ. FIELDS EXCEEDS MAXIMUM ØF ',I3)
0093              IERFLG=1
```

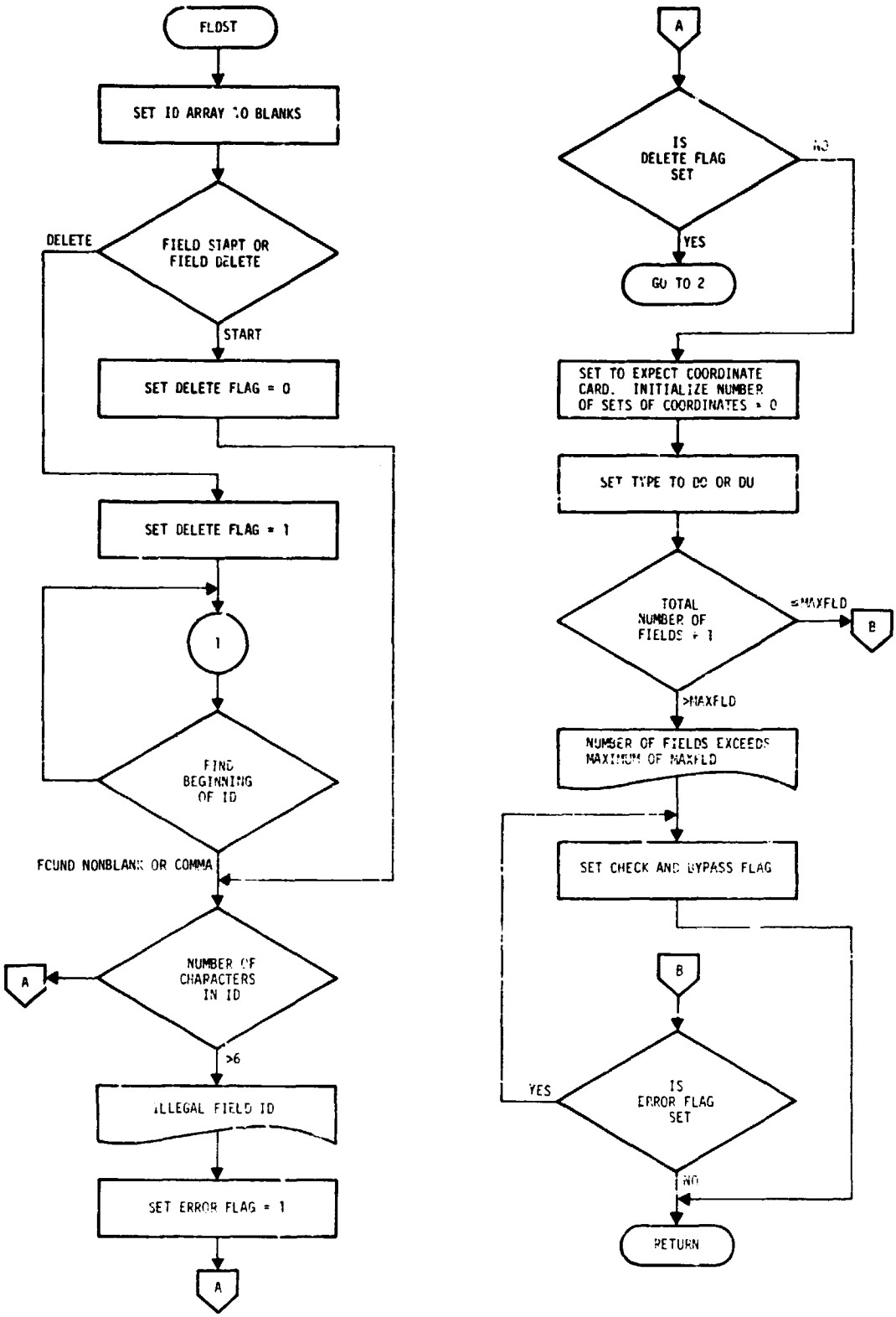
ORIGINAL PAGE IS
OF POOR QUALITY

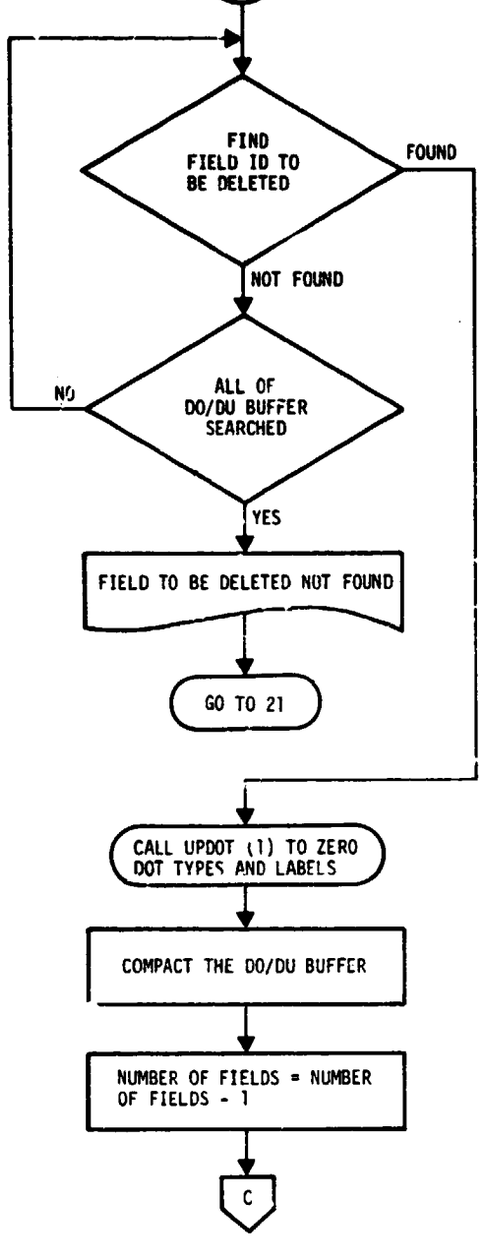
```
FLDUPD.FTN      /TR:BLØCKS/WR
0094            NEWL=1
0095         46 CØNTINUE
0096            KCKBP=1
0097            GØ TØ 500
0098         48 CØNTINUE
0099            IF (IERFLG .NE. 0) GØ TØ 46
0100              IF (KCKBP .NE. 0) GØ TØ 500
                C     STØRE FIELD ID IN DØ/DU BUFFER
0101            N=NFLD*NRDØUE
0102            DØ 50 I=1,6
0103              LDØDU(I+N)=IDFLD(I)
0104         50   CØNTINUE
                C       SET DØ(=1), DU(=2) TYPE INDICATØR
0105              IT=1
0106              IF (ITYPE .EQ. 5) IT=2
0107              LDØDU(N+7)=IT
0108            GØ TØ 500
0109         90 CØNTINUE
                C     DELETE FIELD - FIND ID IN DØ/DL BUFFER
0110            DØ 94 I=1,NBDØU,NBDØUE
0111            IF (LDØDU(I) .NE. IDFLD(1)) GØ TØ 94
0112            IF (LDØDU(I+1) .NE. IDFLD(2)) GØ TØ 94
0113            IF (LDØDU(I+2) .NE. IDFLD(3)) GØ TØ 94
0114            IF (LDØDU(I+3) .NE. IDFLD(4)) GØ TØ 94
0115            IF (LDØDU(I+4) .NE. IDFLD(5)) GØ TØ 94
0116            IF (LDØDU(I+5) .NE. IDFLD(6)) GØ TØ 94
                C     FØUND IT
0117            GØ TØ 100
0118         94 CØNTINUE
                C     FIELD NØT FØUND
0119            IF (NEWL .NE. 0) WRITE (7,5000)
0120            WRITE (7,1094)
0121       1094 FØRMAT (1H+,80X,'FIELD TØ BE DELETED NØT FØUND')
0122            NEWL=1
0123            GØ TØ 490
0124        100 CØNTINUE
0125            IXDØU=1
                C     UPDATE DØT DATA FILE IF IT EXISTS
0126              CALL UPDØT(2)
                C     CØMPACT DØ/DU BUFFER WHEN NECESSARY
                C     IS DELETED FIELD LAST ØNE IN BUFFER
0127            IF ((NFLD-1)*NRDØUE+1 .EQ. IXDØU) GØ TØ 104
0128            M=IXDØU+NBDØUE
0129            N=NFLD*NRDØUE
0130            J=IXDØU
0131            DØ 102 I=M,N
0132            LDØDU(J)=LDØDU(I)
0133            J=J+1
0134        102 CØNTINUE
0135        104 CØNTINUE
0136            NFLD=NFLD-1
0137            IACTV=1
0138            IF (NFLD .GT. 0) GØ TØ 490
0139            IACDU=1
                C     FØRM DØ/DU FILE NAME
0140            ENCØDE (4,1300,IDA) ID
```

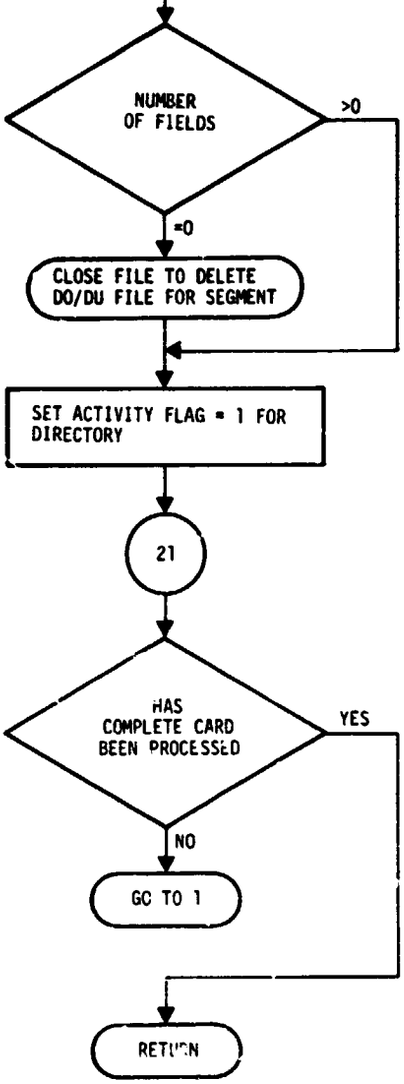
FLDUPD,FTN /TR:RL0CKS/WR

```
0141      1300 FØRMAT (4A1)
          C         STØRE ALPHANUMERIC SEGMENT ID
0142            NAMFIL(14)=IDA(1)
0143            NAMFIL(15)=IDA(2)
0144            NAMFIL(16)=IDA(3)
0145            NAMFIL(17)=IDA(4)
0146            J=18
0147            K=5
0148            DØ 2520 I=1,9
0149            NAMFIL(J)=NAMFLD(I)
0150            NAMFIL(K)=IUIC(I)
0151            J=J+1
0152            K=K+1
0153       2520 CØNTINUE
0154              IACTV=0
0155              CLØSE (UNIT=LU1,DISPØSE='DELETE',ERR=109)
0156              GØ TØ 110
0157        109 CØNTINUE
0158              WRITE (7,1109) (NAMFIL(I),I=1,26)
0159       1109 FØRMAT (81X,'ERRØR-FILE DELETE',/,81X,26A1)
0160        110 CØNTINUE
          C         ZERØ DATE IN DIRECTØRY FILE BUFFER
0161              N=IBDIR-1
0162            DØ 256 I=1,4
0163              LDIR(I+N)=IZERØ
0164        256 CØNTINUE
0165        490 CØNTINUE
0166            IB=IB+5
0167            IF (IB .LT. 72) GØ TØ 12
0168        500 CØNTINUE
0169            RETURN
0170            END
```

2.4 SUBROUTINE FLDST







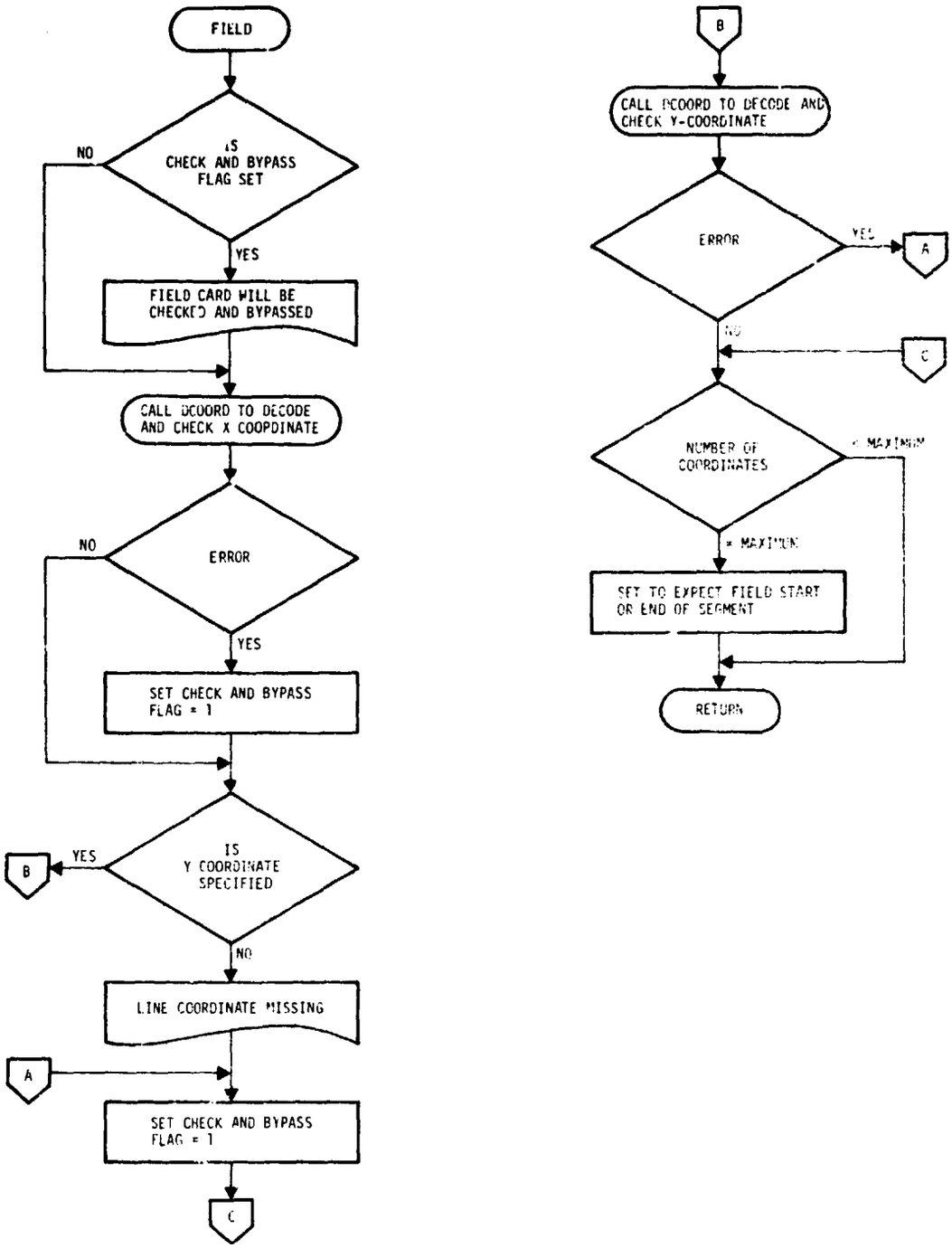
ORIGINAL PAGE IS
OF POOR QUALITY

2.5 SUBROUTINE FIELD

```
FØRTRAN IV+PLUS V02-04          08:47:54    11-MAY-77          PAGE 25
FLDUPD.FTN      /TR:RL@CKS/WR
0001            SUBR@UTINE FIELD
        CCC
        C
        C         FIELD CARD WAS READ
        C
        CCC
0002            C@MM@N /C@M1/ IACTV,IACDI,NEWL,INDEXP(4),IERFLG,KCKBP,
               1 IDLTE,NDEXR,ID@DU
0003            BYTE ID@DU
0004            C@MM@N /C@M2/ ID,IDFLD(6),ICARD(80),ITYPE
0005            BYTE IDFLD,ICARD,ITYPE
0006            C@MM@N /C@M3/ NFLD,NC@@RD
0007            C@MM@N /FILES/ IN,I@UT,LU1,LU2,LU3,LU4,M@UNT
0008              INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0009 *          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     *          1,MAXV=11,ND@TS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *          2N@SPND=6,NPDTWD=10
0010            C@MM@N /FLDC@M/ IVRTC(2,MAXV)
0011            BYTE MINUS
0012            DATA MINUS /1H-/
0013              NVT@T=MAXV
0014            IF (KCKBP .EQ. 0) G@ T@ 90
0015            WRITE (7,1090)
0016         1090 F@RMAT (A1X,'FIELD CARD WILL BE CHECKED AND BYPASSED')
0017           90 C@NTINUE
0018            NC@@RD=NC@@RD+1
        C         DEC@DE AND CHECK PIXEL
0019            ICNT=2
0020            CALL DC@@RD (ICARD(2),NC@@RD,1,IERR)
0021            IF (IERR .NE. 0) G@ T@ 190
0022          130 C@NTINUE
0023            ICNT=9
        C         CHECK F@R LINE CARD
0024            IF (ICARD(8) .NE. MINUS) G@ T@ 900
0025            ICNT=9
0026            CALL DC@@RD (ICARD(9),NC@@RD,2,IERR)
0027            IF (IERR .NE. 0) G@ T@ 190
0028          140 C@NTINUE
0029            IF (NC@@RD .LT. NVT@T) G@ T@ 500
        C         SET T@ EXPECT FIELD START CARD @R END @F SEGMENT
0030            D@ 148 I=1,4
0031            INDEXP(I)=0
0032          148 C@NTINUE
0033            INDEXP(2)=1
0034              INDEXP(3)=2
0035            INDEXP(4)=1
0036            G@ T@ 500
0037          190 C@NTINUE
0038            KCKBP=1
0039          200 C@NTINUE
0040            IF (ICNT .LT. 9) G@ T@ 130
0041              G@ T@ 140
0042          500 C@NTINUE
0043            RETURN
0044          900 C@NTINUE
0045            IF (IXY .EQ. 1) IERR=0
```

```
FØRTRAN IV+PLUS V02+04          08147154    11+MAY+77          PAGE 26
FLDUPD,FTN      /TR:BL@CKS/WR
0046            WRITE (7,1900)
0047      1900  FØRMAT (81X,'LINE C@@RDINATE M:SSING')
0048            G@ T@ 190
0049            END
```

2.5 SUBROUTINE FIELD

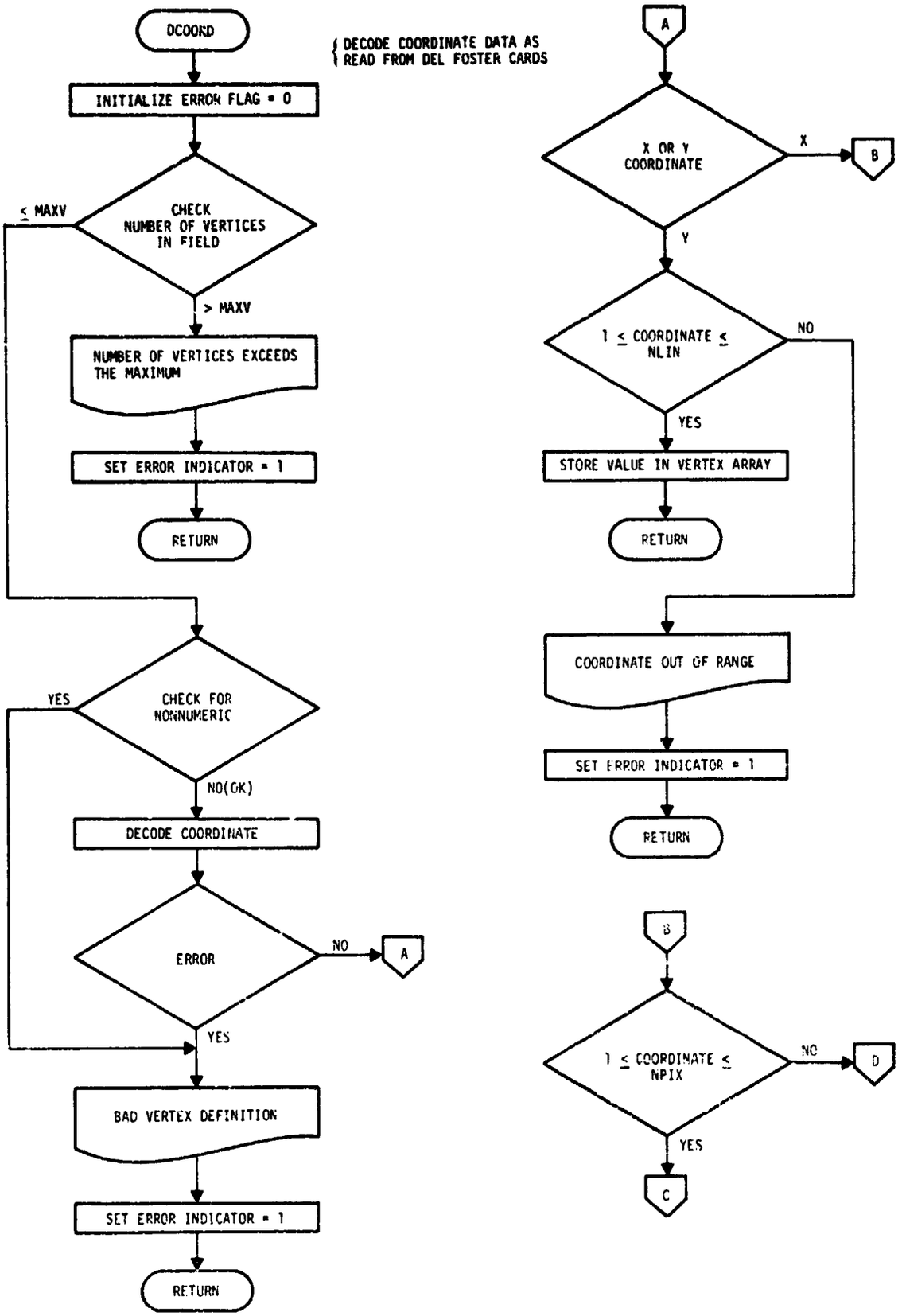


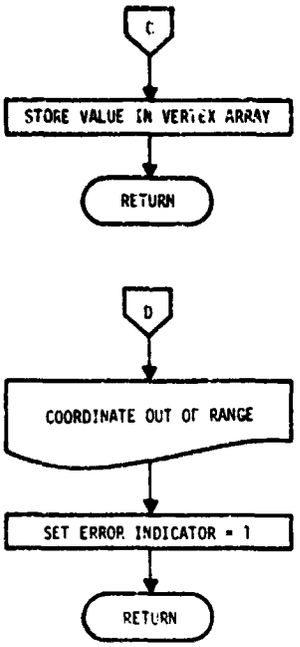
2.6 SUBROUTINE DCOORD

```
FØRTRAN IV-PLUS V02-04          08148103    11-MAY-77          PAGE 29
FLDUPD.FTN      /TR:BLØCKS/WR
0001            SUBRØUTINE DC0ØRD (IBUF,INX, IXY,IERR)
                CCC
                C
                C         DECØDE C0ØRDINATE DATA ØN CARD
                C
                CCC
0002            C0MMØN /FILES/ IN,IØUT,LU1,LU2,LU3,LU4,MØUNT
                C         MAXP = MAXIMUM PIXEL VALUE
                C         MAXL = MAXIMUM LINE VALUE
0003              INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0004 *          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     *         1,MAXV=11,NDØTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *         2NØSPWD=6,NØDTWD=10
0005            C0MMØN /FLDC0M/ IVRTC(2,MAXV)
                C         IBUF = C0NTAINS DATA TØ BE DECØDED
                C         INX = INDEX ØF VERTICE ARRAY (NØ. ØF C0ØRDINATES)
                C         IXY = 1 PIXEL
                C         IXY = 2 LINE
                C         IERR = 0 NØ ERRØR
                C         IERR = 1 ERRØR
0006            BYTE IBUF(1)
0007            BYTE ID0,ID9
0008            DATA ID0/'0'/, ID9/'9'/
0009              MAXP=NPIX
0010              MAXL=NLIN
0011              NVTØT=MAXV
0012            IERR=0
                C         CHECK INDEX
0013            IF (INX .GT. NVTØT) GØ TØ 901
0014            DØ 100 I=1,6
0015            IF (IBUF(I) .LT. ID0 .ØR. IBUF(I) .GT. ID9)
               1  GØ TØ 903
0016        100 C0NTINUE
0017            DECØDE (6,1100,IBUF(1),   ERR=903) IC0ØR
0018       1100 FØRMAT (I6)
                C         CHECK RANGE ØF C0ØRDINATE
0019            MAX=MAXP
0020            IF (IXY .EQ. 2) MAX=MAXL
0021            IF (IC0ØR .LT. 1 .ØR. IC0ØR .GT. MAX) GØ TØ 902
0022            IVRTC(IXY,INX)=IC0ØR
0023        500 C0NTINUE
0024            RETURN
0025        901 C0NTINUE
0026            WRITE (7,1901)
0027       1901 FØRMAT(1X,80X,'NØ. VERTICES EXCEEDS MAXIMUM!')
0028            IERR=1
0029            GØ TØ 500
0030        902 C0NTINUE
0031            WRITE (7,1902) IC0ØR
0032       1902 FØRMAT (1X,80X,'C0ØRDINATE ØUT ØF RANGE!',I5)
0033            IERR=1
0034            GØ TØ 500
0035        903   C0NTINUE
0036              WRITE (7,1903)
0037       1903   FØRMAT (1X,80X,'BAD VERTEX DEFINITIØN!')
0038              IERR=1
```

```
FØRTRAN IV-PLUS V02-04          08148103    11-MAY-77          PAGE 30
FLDUPD.FTN      /TRIBLOCKS/WR
0039            GØ TØ 500
0040        END
```

2.6 SUBROUTINE DCOORD




C
STORE VALUE IN VERTEX ARRAY
RETURN
D
COORDINATE OUT OF RANGE
SET ERROR INDICATOR = 1
RETURN

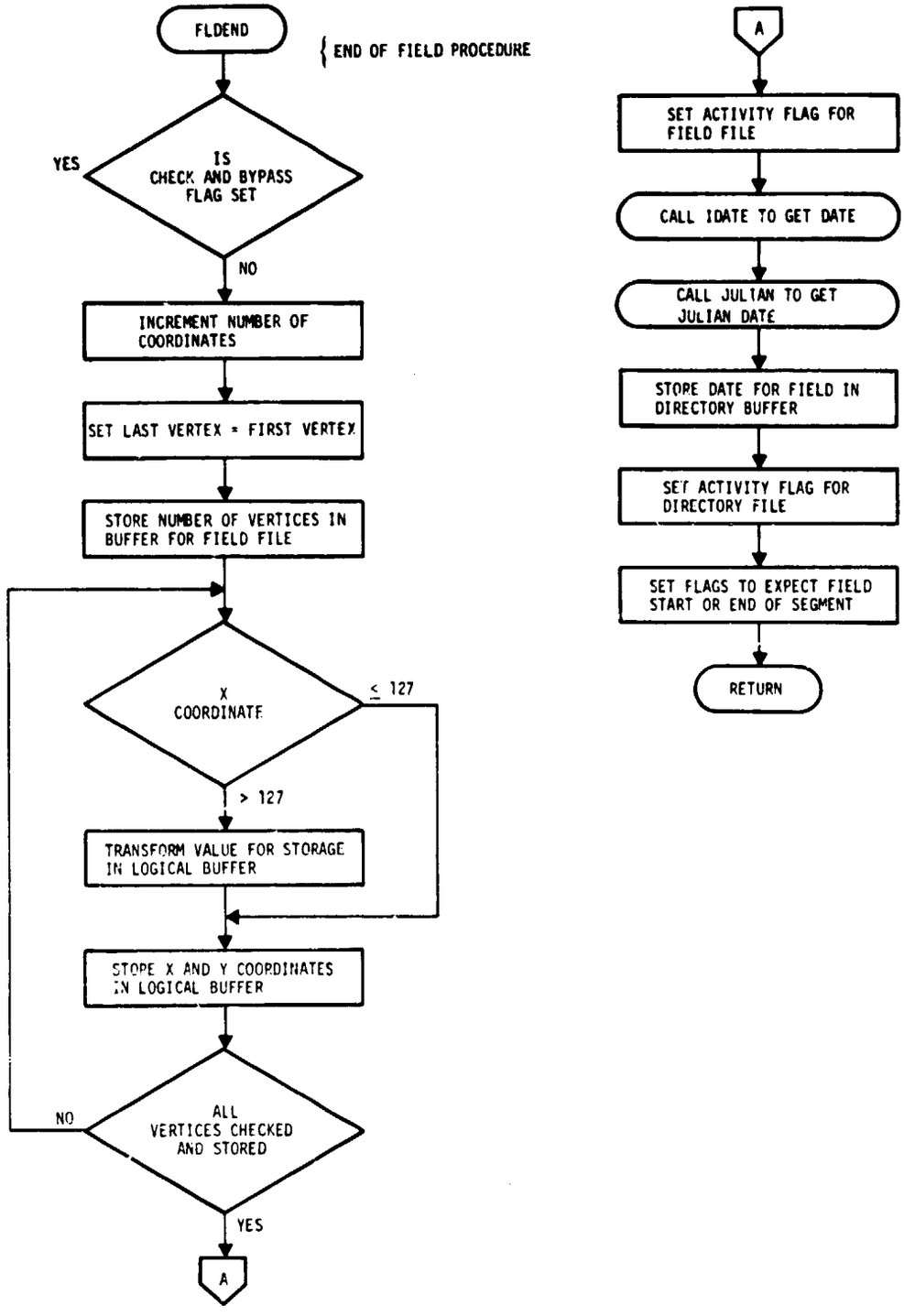
2.7 SUBROUTINE FLDEND

```
FORTRAN IV+PLUS V02-04          08148111    11-MAY-77          PAGE 32
FLDUPD.FTN      /TR:BLOCKS/WR
0001            SUBROUTINE FLDEND
                CCC
                C
                C       FIELD END (FLDEND) CARD READ
                C
                CCC
0002            COMMON /COM1/ IACTV,IACDI,NEWL,INDEXP(4),IERFLG,KCKBP,
               1 IDLTE,NDEXR,IDODU
0003             BYTE IDODU
0004             COMMON /COM2/ ID,IDFLD(6),ICARD(80),ITYPE
0005             BYTE IDFLD,ICARD,ITYPE
0006             COMMON /FILES/ IN,IOUT,LU1,LU2,LU3,LU4,MOUNT
0007               INCLUDE 'SYS[300,3]CAMSPARAM.INC'
0008 +           PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=190,NLIN=117,MAXFLD=50
     +           1,MAXV=11,NDOTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     +           2NOSPWD=6,NODTWD=10
0009             COMMON /COM3/ NFLD,NCORD
0010             PARAMETER NBDR=512,NBTBL=400,NBDOL=1450
0011               PARAMETER NWDR=256
0012             PARAMETER NBDOUE=30
0013             COMMON /BUFRS/ IXDIR,IBDIR,IXINX,IXDOU,
                1  LDTBL(NBTBL),LDIR(NBDR),LDODU(NBDOU)
0014             BYTE LDIR,LDODU
0015               DIMENSION IDIR(NWDR)
0016               EQUIVALENCE (IDIR,LDIR)
0017             COMMON /FLDCOM/ IVRTC(2,MAXV)
0018               PARAMETER NBDOT=10868,NBPDOT=52
0019               COMMON /COMDOT/ IXDOT,LDOT(NBDOT)
0020               BYTE LDOT
0021             BYTE KTXY(2)
0022             DATA KTXY /1HX,1HY/
0023               NVTOT=MAXV
0024               NVTOT=MAXV
0025               IF (KCKBP .NE. 0) GO TO 35
0026             NCORD=NCORD+1
0027             IVRTC(1,NCORD)=IVRTC(1,1)
0028             IVRTC(2,NCORD)=IVRTC(2,1)
0029             NC2=NCORD+2
0030             N=NFLD+NBDOUE+8
0031             LDODU(N)=NCORD
0032             J=1
0033             DO 30 I=1,NC2,2
0034               IV=IVRTC(1,J)
0035               IL=IV
0036               IF (IV .LE. 127) GO TO 28
0037               IL=-((.NOT. IV)+1)
0038          28 CONTINUE
0039               LDODU(I+N)=IL
0040               LDODU(I+N+1)=IVRTC(2,J)
0041               J=J+1
0042          30 CONTINUE
0043               NFLD=NFLD+1
              C       SET ACTIVITY FLAG
0044          35 CONTINUE
              C       SET DATE IN DIRECTORY
0045               IACTV=1
```

ORIGINAL PAGE IS OF POOR QUALITY

```
FLDUPD,FTN      /TR:BLØCKS/WR
0046            N=IBDIR-1
0047            CALL IDATE (MØ,IDA,IYR)
0048            CALL JULIAN (IYR,MØ,IDA,JULIØ)
0049            N=IBDIR/2
0050            IF (MØD(IBDIR,2) .NE. 0) N=N+1
0051            IDIR(N)=IYR
0052            IDIR(N+1)=JULIØ
0053          IACDI=1
0054    35      CØNTINUE
        C         SET TØ EXPECT FIELD START ØR SEGEND
0055          DØ  40 I=1,4
0056          INDEXP(I)=0
0057       40 CØNTINUE
0058          INDEXP(2)=1
0059          INDEXP(4)=1
0060      500 CØNTINUE
0061          RETURN
0062          END
```

2.7 SUBROUTINE FLDEND

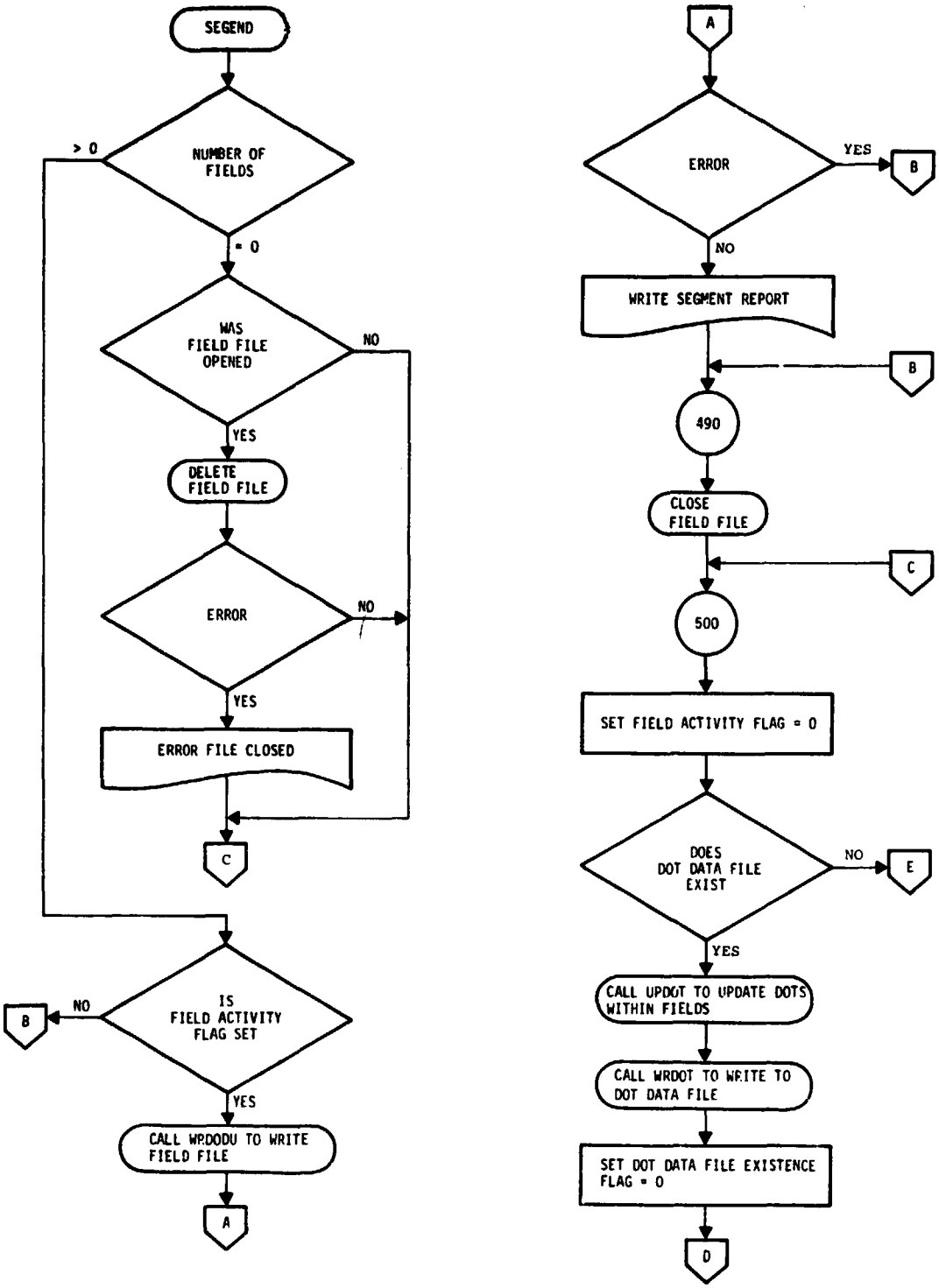


2.8 SUBROUTINE SEGEND

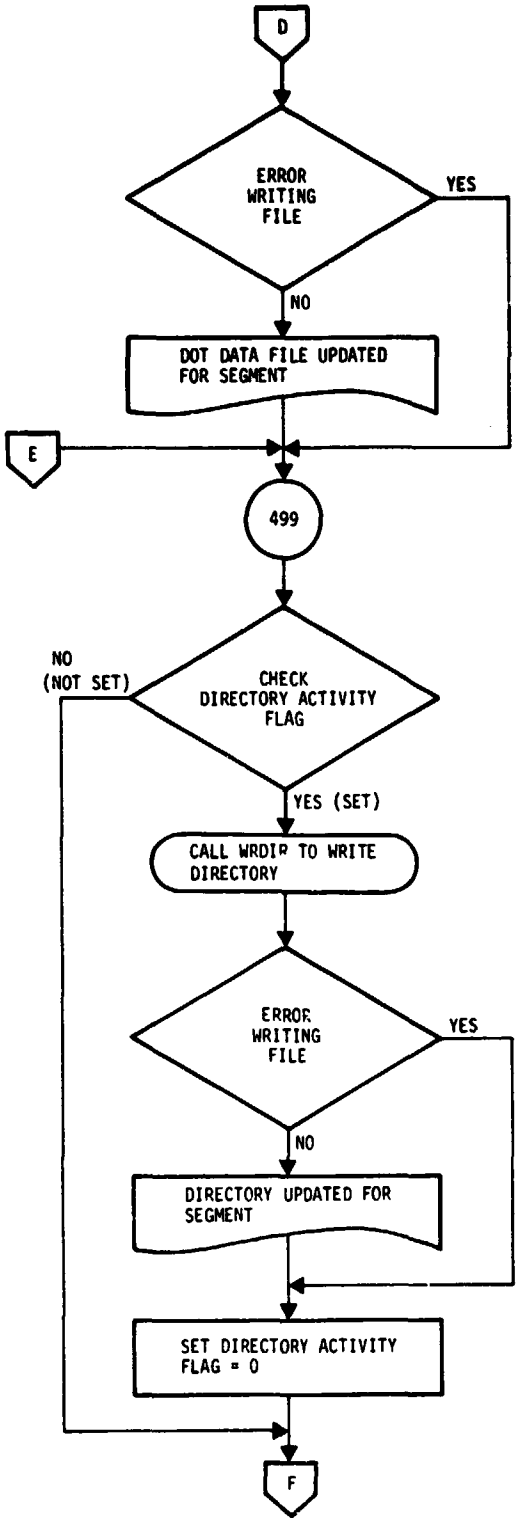
```
FØRTRAN IV+PLUS V02-04          08148122    11-MAY-77          PAGE 36
FLDUPD.FTN      /TRIBLØCKS/WR
0001            SUBRØUTINE \SEGEND
                CCC
                C
                C       END ØF SEGMENT CARD (SEGEND) CARD READ
                C
                CCC
0002            CØMMØN /C0M1/ IACTV,IACDI,NEWL,INDEXP(4),IERFLG,KCKBP,
                1 IDLTF,NDEXR,ID0DU
0003            BYTE ID0DU
0004            CØMMØN /C0M2/ ID,IDFLD(6),ICARD(80),ITYPE
0005            BYTE IDFLD,ICARD,ITYPE
0006            PARAMETER NBDR=512,NBTBL=400,NBD0L=1450
0007            PARAMETER NBD0UE=30
0008            CØMMØN /BUFRS/ IXDIR,IBDIR,IXINX,IXD0U,
                1  LDTBL(NBTBL),LDIR(NBDR),LD0DU(NBD0U)
0009            BYTE LDIR,LD0DU
0010            CØMMØN /FILES/ IN,IØUT,LU1,LU2,LU3,LU4,MØUNT
0011            CØMMØN /C0M3/ NFLD,NC00RD
0012              INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0013 *          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     *          1,MAXV=11,NDØTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *          2NØSPWD=6,NADTWD=10
0014              PARAMETER NBD0T=10865,NBD0AT=52
0015              CØMMØN /C0MD0T/IXD0T,LD0T(NBD0T)
0016              DIMENSIØN IX(MAXV),IY(MAXV)
0017              BYTE LD0T
0018              BYTE L1
0019              BYTE LD0,LDU
0020              DATA LD0/1HD/, LDU/1HU/
0021         15 CØNTINUE
             C       IS ACTIVITY FLAG SET
0022              IF (NFLD .EQ. 0) GØ TØ 503
0023            IF (IACTV .EQ. 0) GØ TØ 490
             C       WRITE TØ D0/DU FILE
0024              CALL WRD0DU (ID,IERR)
0025              IF (IERR .NE. 0) GØ TØ 490
0026            WRITE (7,1015)
0027         1015 FØRMAT (81X,'SEGMENT WRITTEN TØ D0/DU FILE')
0028              WRITE (7,1000) ID
0029         1000 FØRMAT (1X,'SEGMENT ',I5)
0030              N=1
0031              DØ 9004 J=1,NFLD
0032              NV=LD0DU(N+7)-1
0033              NN=NV+2
0034              LAB=LD0
0035              IF (LD0DU(N+6) .NE. -1) LAB=LDU
0036              N7=N+7
0037              NN1=N+5
0038              K=1
0039              DØ 20 I=1,NN,2
0040              N7I=N7+I
0041              L1=LD0DU(N7I+1)
0042              IY(K)=IBYTE(0,L1)
0043              L1=LD0DU(N7I)
0044              IX(K)=IBYTE(0,L1)
0045              K=K+1
```

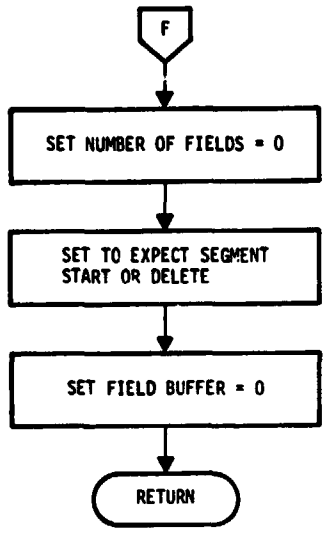
```
FLDUPD.FTN      /TRIPLØCKS/WR
0046         20 CØNTINUE
0047            WRITE (7,1010) (LDØDU(I),I=N,NN1),LAB,
                1  (IY(I),I=1,NV)
0048            WRITE (7,1011) (IX(I),I=1,NV)
0049       1010 FØRMAT (1X,6A1,9X,A1,4X,'LINE ',I8)
0050       1011 FØRMAT (21X,'PIXEL',I8,/)
0051            N=J+NBDØUE+1
0052       9004 CØNTINUE
0053        490 CØNTINUE
           C         RESET  ACTIVITY FLAG
0054            CLØSE(UNIT=LU1)
0055        500 CØNTINUE
0056            IACTV=0
0057            IF (IXDØT .EQ. 0) GØ TØ 499
0058            CALL UPDØT(2)
0059            CALL WRDØT (ID,IERR)
0060            IXDØT=0
0061            IF (IERR .EQ. 0) WRITE (7,1490)
0062       1490 FØRMAT (81X,'DØT DATA FILE UPDATED FØR SEGMENT')
           C
0063        499 CØNTINUE
           C         WRITE TØ DIRECTØRY IF NECESSARY
0064            IF (IACDI .EQ. 0) GØ TØ 501
0065            CALL WRDIR (ID,IERR)
0066            IF (IERR .EQ. 0) WRITE (7,1500)
0067       1500 FØRMAT (81X,'DIRECTØRY UPDATED FØR SEGMENT')
0068            IACDI=0
0069        501 CØNTINUE
0070            NFLD=0
           C          SET TØ EXPECT SEG START ØR DELETE
0071            INDEXP(1)=1
0072            DØ 502 I=2,4
0073            INDEXP(I)=0
0074        502 CØNTINUE
0075            RETURN
0076        503 CØNTINUE
           C         WAS DØ/DU FILE ØPENED
0077            IF (KCKBP .LT. 0) GØ TØ 500
0078            CLØSE (UNIT=LU1,DISPØSE='DELETE',ERR=504)
0079            GØ TØ 500
0080        504 CØNTINUE
0081            WRITE (7,1504)
0082       1504 FØRMAT (1X,'ERRØR•FILE CLØSE')
0083            GØ TØ 500
0084            END
```

2.8 SUBROUTINE SEGEND



C-2

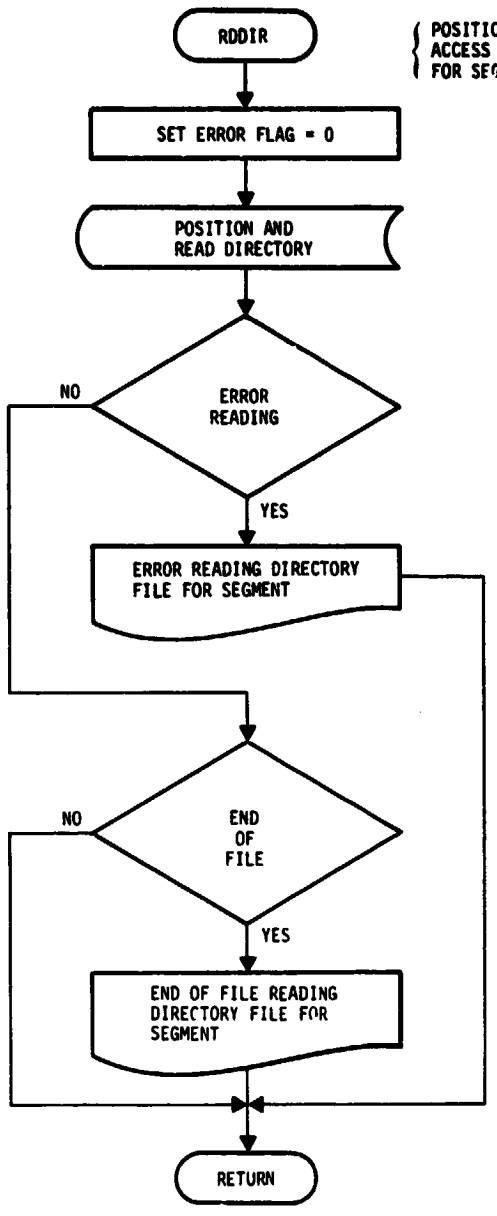

D
ERROR WRITING FILE
YES
NO
DOT DATA FILE UPDATED FOR SEGMENT
E
499
NO (NOT SET)
CHECK DIRECTORY ACTIVITY FLAG
YES (SET)
CALL WRDIR TO WRITE DIRECTORY
ERROR WRITING FILE
YES
NO
DIRECTORY UPDATED FOR SEGMENT
SET DIRECTORY ACTIVITY FLAG = 0
F


F
SET NUMBER OF FIELDS = 0
SET TO EXPECT SEGMENT START OR DELETE
SET FIELD BUFFER = 0
RETURN

2.9 SUBROUTINE RDDIR

```
FØRTRAN IV+PLUS V02-04          08:48:57    11-MAY-77          PAGE 50
FLDUPD.FTN      /TRIBLØCKS/WR
0001            SUBRØUTINE RDDIR(ID,IERR)
                CCC
                C
                C         READ DIRECTØRY FILE FØR SEGMENT
                C
                CCC
0002                COMMØN /CØM1/ IACTV,IACDI,NEWL,INDEXP(4),IERFLG,KCKBP,
                   1 IDLTE,NDEXR,IDØDU
0003                BYTE IDØDU
0004                PARAMETER NBDR=512,NBTBL=400,NBDØU=1450
0005                  PARAMETER NWDR=256
0006                PARAMETER NBDØUE=30
0007                CØMMØN /BUFRS/ IXDIR,IBDIR,IXINX,IXDØU,
                   1  LDTBL(NBTBL),LDIR(NBDR),LDØDU(NBDØU)
0008                BYTE LDIR,LDØDU
0009                  DIMENSIØN IDIR(NWDR)
0010                  EQUIVALENCE (IDIR(1),LDIR)
0011                CØMMØN /FILES/ IN,IØUT,LU1,LU2,LU3,LU4,MØUNT
0012                CØMMØN /MSGRW/ MSG1(6),MSG2(12),MSGR(9),
                   1  MSGW(9),MSGDI(9),MSGDØU(5),MSGDØT(8),MSGND(17)
0013                BYTE MSG1,MSG2,MSGR,MSGW,MSGDI,MSGDØU,MSGDØT,MSGND
0014                IERR=0
0015                READ (LU2'IXDIR,ERR=100,END=102) IDIR
0016             500 CØNTINUE
0017            9000 FØRMAT (40X,'DIR ',317)
0018                 RETURN
0019             100 CØNTINUE
0020                 IERR=1
0021                 WRITE (7,2000) MSG1,MSGR,MSGDI,MSGND,ID
0022            2000 FØRMAT (1X,41A1,I6)
0023                 GØ TØ 500
0024             102 CØNTINUE
0025                 IERR=2
0026                 WRITE (7,2001) MSG2,MSGR,MSGDI,MSGND,ID
0027            2001 FØRMAT (1X,47A1,I6)
0028                 GØ TØ 500
0029                 END
```

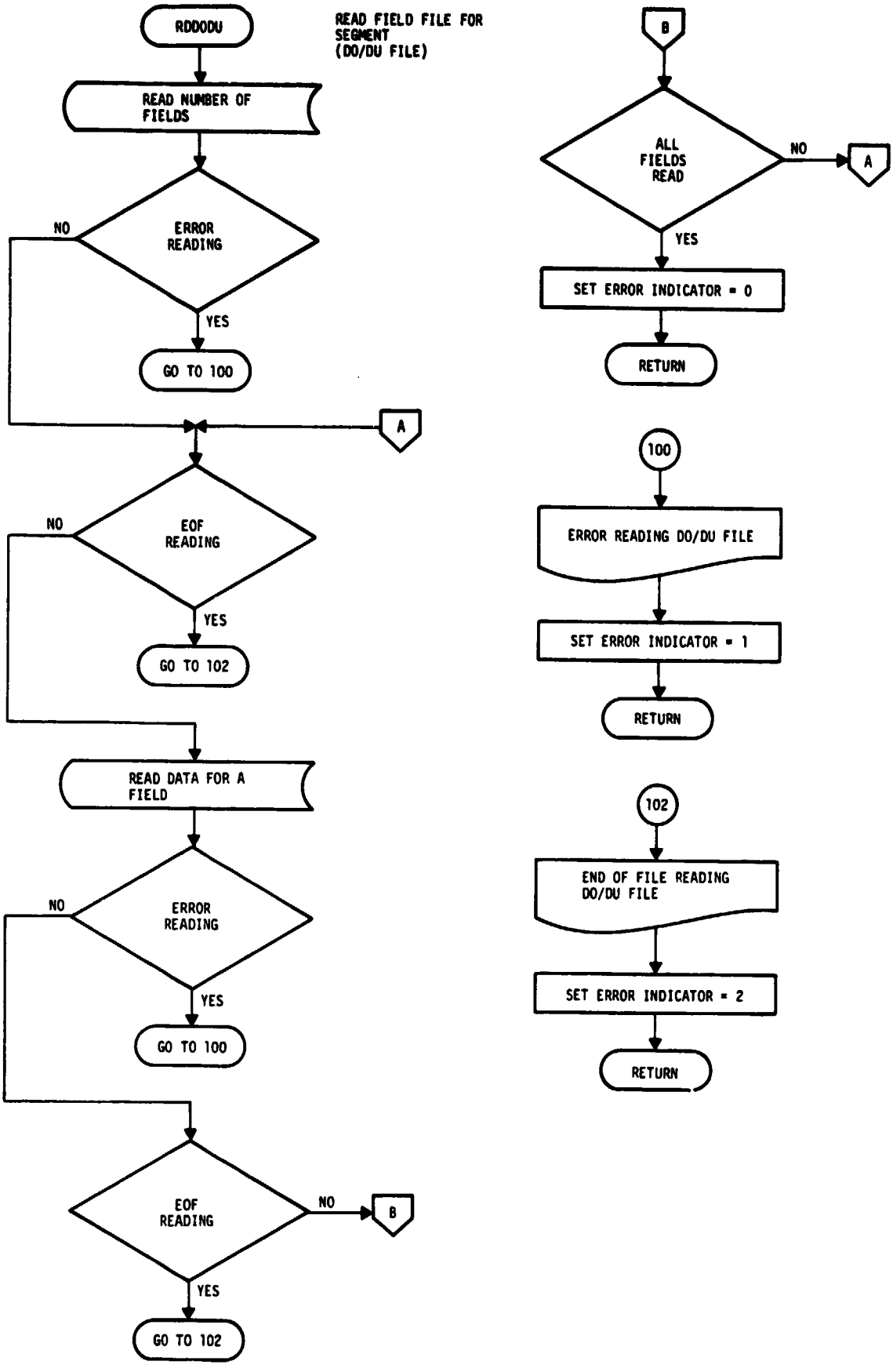
2.9 SUBROUTINE RDDIR



2.10 SUBROUTINE RDDODU

```
FØRTRAN IV+PLUS V02-04          08:49:01    11-MAY-77          PAGE 53
FLDUPD.FTN      /TRIBLØCKS/WR
0001            SUBRØUTINE RDDØDU(ID,IERR)
                CCC
                C
                C         READ DØ/DU FILE FØR SEGMENT
                C
                CCC
0002            CØMMØN /CØM1/ IACTV,IACDI,NEWL,INDEXP(4),IERFLG,KCKBP,
                1 IDLTE,NDEXR,IDØDU
0003            BYTE IDØDU
0004            CØMMØN /CØM3/ NFLD,NCØØRD
0005            PARAMETER NBDR=512,NBTBL=400,NBDØL=1450
0006            PARAMETER NBDØUE=30
0007            CØMMØN /BUFRS/ IXDIR,IBDIR,IXINX,IXDØU,
                1  LDTBL(NBTBL),LDIR(NBDR),LDØDU(NBDØU)
0008            BYTE LDIR,LDØDU
0009            CØMMØN /FILES/ IN,IØUT,LU1,LU2,LU3,LU4,MOUNT
0010            CØMMØN /MSGRW/ MSG1(6),MSG2(12),MSGR(9),
                1  MSGW(9),MSGDI(9),MSGDØU(5),MSGDØT(8),MSGND(17)
0011            BYTE MSG1,MSG2,MSGR,MSGW,MSGDI,MSGDØU,MSGDØT,MSGND
                C         READ NØ. ØF FILES
0012            READ (LU1,ERR=100,END=102) NFLD
                C         READ REST ØF FILE
0013              N=1
0014              DØ 90 J=1,NFLD
0015              NN=N+NBDØUE-1
0016              READ (LU1,ERR=100,END=102) (LDØDU(I),I=N,NN)
0017              N=N+NBDØUE
0018           90 CØNTINUE
0019              IERR=0
0020          500 CØNTINUE
0021              RETURN
0022          100 CØNTINUE
0023              WRITE (7,2000) MSG1,MSGR,MSGDØU,MSGND,ID
0024         2000 FØRMAT (1X,37A1,I6)
0025              IERR=1
0026              GØ TØ 500
0027          102 CØNTINUE
0028              IERR=2
0029              WRITE (7,2001) MSG2,MSGR,MSGDØU,MSGND,ID
0030         2001 FØRMAT (1X,43A1,I6)
0031              GØ TØ 500
0032              END
```

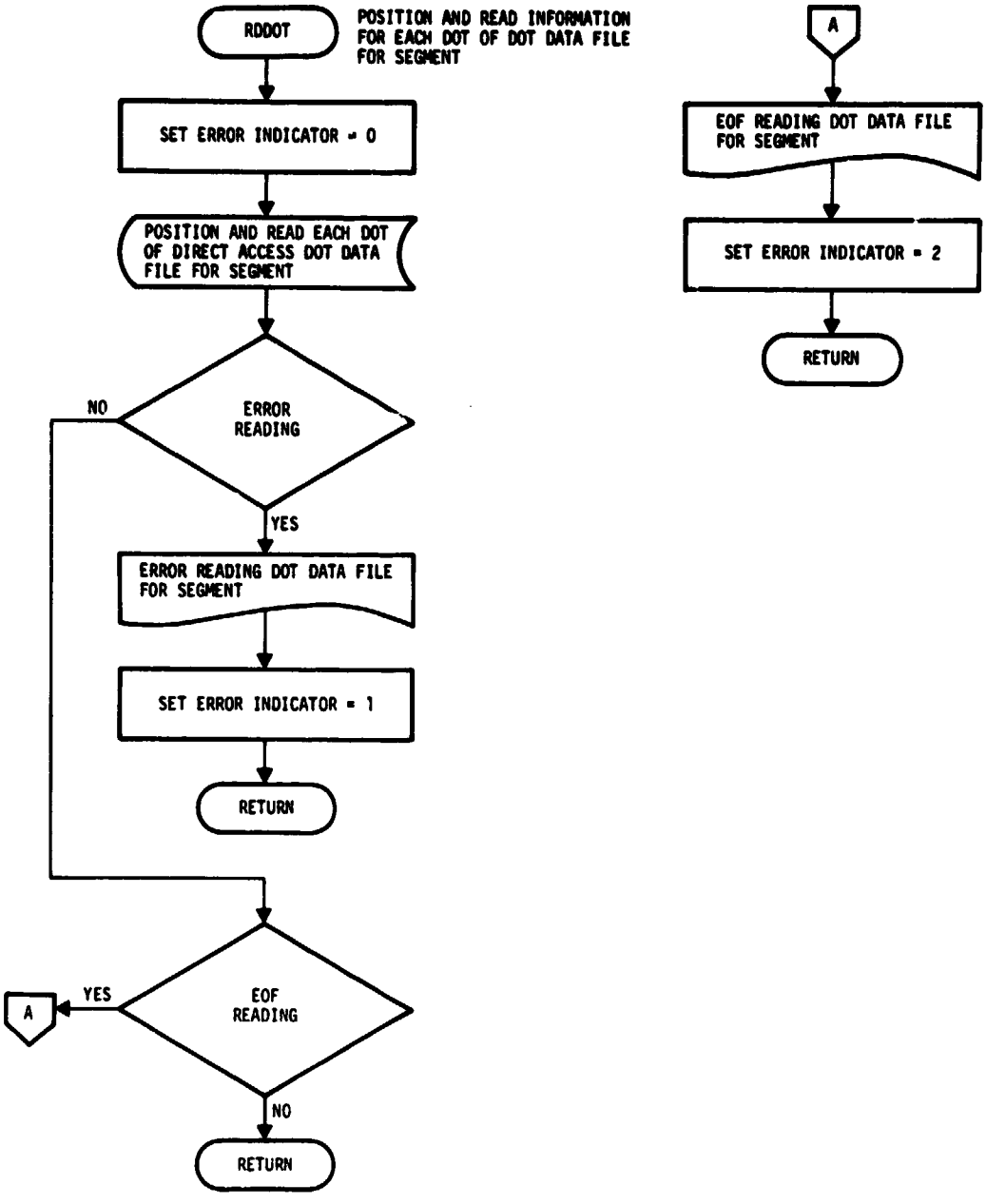
2.10 SUBROUTINE RDDODU



2.11 SUBROUTINE RDDOT

```
FØRTRAN IV+PLUS V02+04          08:49:12    11+MAY+77          PAGE 59
FLDUPD,FTN      /TR:BL0CKS/WR
0001            SUBR0UTINE RDD0T (ID,IERR)
                CCC
                C
                C         READ D0T DATA FILE
                C
                CCC
0002              INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0003 *          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     *         1,MAXV=11,ND0TS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *         2N0SPWD=6,N?DTWD=10
0004            PARAMETER NBD0T=10968,NBPD?T=52
0005            C0MMON /C0MD0T/ IXD0T,LD?T(NPD?T)
0006              BYTE LD0T
0007            C0MMON /FILES/ IN,10UT,LU1,LU2,LU3,LU4,M0UNT
0008            C0MMON /MSGRW/ MSG1(6),MSG2(12),MSGR(9),
                1  MSGW(9),MSGDI(9),MSGD0U(5),MSGD0T(8),MSGND(17)
0009            BYTE MSG1,MSG2,MSGR,MSGW,MSGDI,MSGD0U,MSGD0T,MSGND
0010            IERR=0
0011            J=1
0012            D0 10 I=1,ND0TS
0013            JPN=J+NBPD?T-1
0014            READ (LU3'I,ERR=900,END=902) (LD0T(K),K=J,JPN)
0015            J=J+NBPD?T
0016         10 C0NTINUE
0017        500 C0NTINUE
0018              J=1
0019              D0 11 K=1,ND?TS,3
0020              J=J+156
0021         11 C0NTINUE
0022            RETURN
0023        900 C0NTINUE
0024            WRITE (7,2000) MSG1,MSGR,MSGD0T,MSGND,ID
0025       2000 F0RMAT (1X,40A1,I6)
0026            IERR=1
0027            G0 T0 500
0028        902 C0NTINUE
0029            WRITE (7,2001) MSG2,MSGR,MSGD0T,MSGND,ID
0030       2001 F0RMAT (1X,46A1,I6)
0031            IERR=2
0032            G0 T0 500
0033            END
```

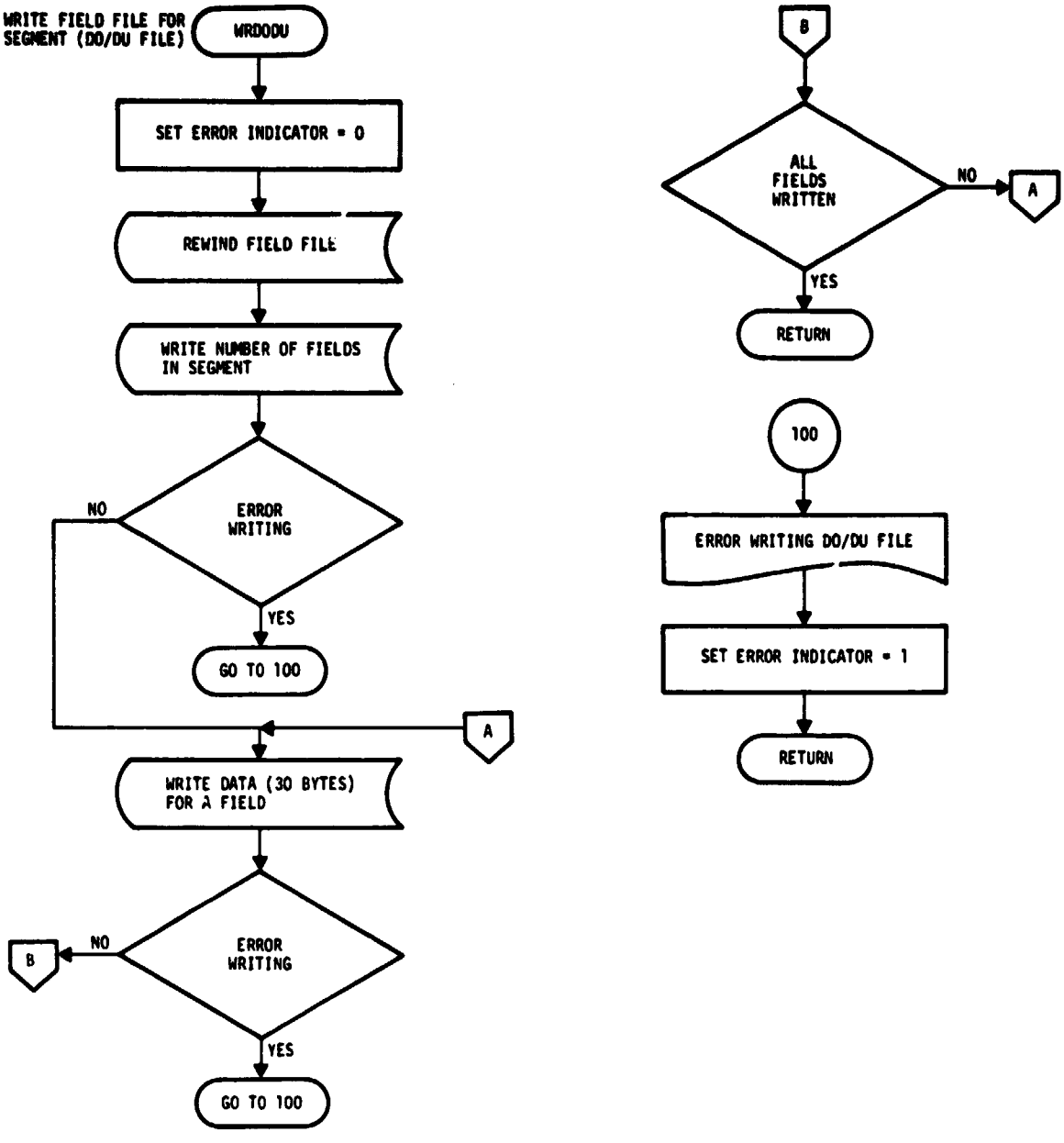
2.11 SUBROUTINE RDDOT



2.12 SUBROUTINE WRDODU

```
FØRTRAN IV+PLUS V02+04          08:49:08    11+MAY+77          PAGE 56
FLDUPD.FTN      /TR:BLØCKS/WR
0001            SUBRØUTINE WRD?DU (ID,IERR)
                CCC
                C
                C         WRITE D?/DU FILE FØR SEGMENT
                C
                CCC
0002            CØMMØN /C0M1/ IACTV,IACDI,NEWL,INDEXP(4),IERFLG,KCKBP,
                1 IDLTE,NDEXR,ID?DU
0003            BYTE ID?DU
0004            CØMMØN /C0M3/ NFLD,NC0?RD
0005            CØMMØN /FILES/ IN,I0UT,LU1,LU2,LU3,LU4,M?UNT
0006            PARAMETER NBDR=512,NBTBL=400,NBD0L=1450
0007            PARAMETER NBD0UE=30
0008            CØMM?N /BUFRS/ IXDIR,IDDIR,IXI??X,IXD0U,
                1  LDTBL(NBTBL),LDIR(NBDR),LD?D?(NBD0U)
0009            BYTE LDIR,LD?DU
0010            CØMMØN /MSGRW/ MSG1(6),MSG2(12),MSGR(9),
                1  MSGW(9),MSGDI(9),MSGD0U(5),MSGD0T(8),MSGND(17)
0011            BYTE MSG1,MSG2,MSGR,MSGW,MSGDI,MSGD0U,MSGD0T,MSGND
0012            IERR=0
0013            REWIND LU2
0014            WRITE (LU1,ERR=100) NFLD
0015              N=1
0016              D0 9? J=1,NFLD
0017              NN=N+NBD?UE-1
0018              WRITE (LU1,ERR=100) (LD0?U(I),I=N,NN)
0019              N=N+NBD0UE
0020           9? CØNTINUE
0021          500 CØNTINUE
0022              RETURN
0023          100 CØNTINUE
0024              IERR=1
0025              WRITE (7,2000) MSG1,MSGW,MSGD0U,MSGND,ID
0026         2000 FØRMAT (1X,37A1,I6)
0027              GØ T? 500
0028              END
```

2.12 SUBROUTINE WRDODU

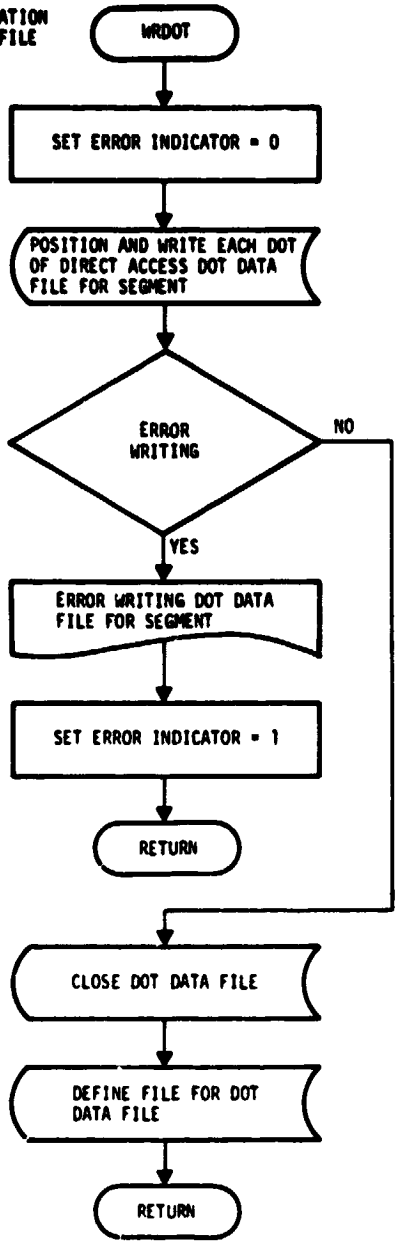


2.13 SUBROUTINE WRDOT

```
FØRTRAN IV+PLUS V02-04          08149119    11-MAY-77          PAGE 61
FLDUPD,FTN      /TRIØLØCKS/WR
0001            SUBRØUTINE WRDØT(ID,IERR)
                CCC
                C
                C         WRITE DØT DATA FILE
                C
                CCC
0002            CØMMØN /FILES/ IN,IØUT,LU1,LU2,LU3,LU4,MØUNT
0003              INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0004 *          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     *         1,MAXV=11,NDØTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *         2NØSPWD=6,NØDTWD=10
0005            PARAMETER NRDØT=10R6R,NØPD'T=52
0006            CØMMØN /CØMDØT/ IXDØT,LDØT(NØLØT)
0007              BYTE LDØT
0008            CØMMØN /MSGRW/ MSG1(6),MSG2(12),MSGR(9),
                1  MSGW(9),MSGDI(9),MSGDØU(5),MSGDØT(8),MSGND(17)
0009            BYTE MSG1,MSG2,MSGR,MSGW,MSGDI,MSGDØU,MSGDØT,MSGND
0010            IERR=0
0011            J=1
0012            DØ 10 I=1,NDØTS
0013            JPN=J+NØPDØT-1
0014            WRITE (LU3'I,ERR=900) (LDØT(K),K=J,JPN)
0015            J=J+NØPDØT
0016         10 CØNTINUE
0017        500 CØNTINUE
0018              J=1
0019              DØ 11 K=1,NDØTS,3
0020              J=J+15A
0021         11 CØNTINUE
0022              CALL CLØSE (LU3)
0023              DEFINE FILE LU3 (NDØTS,2A,U,NEXTD)
0024            RETURN
0025        900 CØNTINUE
0026            WRITE (7,2000) MSG1,MSGW,MSGDØT,MSGND,ID
0027       2000 FØRMAT (1X,40A1,I6)
0028            IERR=1
0029            GØ TØ 500
0030            END
```

2.13 SUBROUTINE WRDOT

POSITION AND WRITE INFORMATION FOR EACH DOT OF DOT DATA FILE FOR SEGMENT

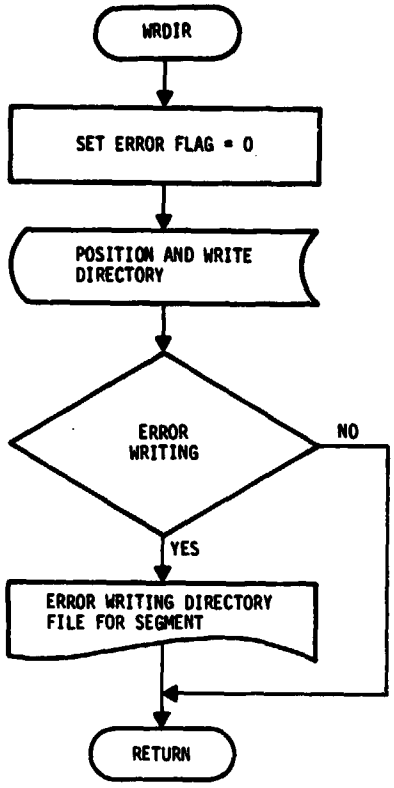


2.14 SUBROUTINE WRDIR

```
FORTRAN IV-PLUS V02-04          08:49:29    11-MAY-77          PAGE 64
FLDUPD.FTN      /TRIBLOCKS/WR
0001            SUBROUTINE WRDIR(ID,IERR)
                CCC
                C
                C          WRITE DIRECTORY FILE FOR SEGMENT
                C
                CCC
0002            COMMON /COM1/ IACTV,IACDI,NEWL,INDEXP(4),IERFLG,KCKBP,
                1 IDLTE,NDEXR,ID0DU
0003            BYTE ID0DU
0004            COMMON /FILES/ IN,IOUT,LU1,LU2,LU3,LU4,MOUNT
0005            PARAMETER NBDR=512,NBTBL=400,NBD0L=1450
0006            PARAMETER NBD0UE=30
0007              PARAMETER NWDR=256
0008            COMMON /BUFRS/ IXDIR,IRDIR,IXIDX,IXD0U,
                1  LDTBL(NBTBL),LDIR(NBDR),LD0DU(NBD0U)
0009            BYTE LDIR,LD0DU
0010              DIMENSION IDIR(NWDR)
0011              EQUIVALENCE (IDIR,LDIR)
0012            COMMON /MSGRW/ MSG1(6),MSG2(12),MSGR(9),
                1  MSGW(9),MSGDI(9),MSGD0U(3),MSGDBT(8),MSGND(17)
0013            BYTE MSG1,MSG2,MSGR,MSGW,MSGDI,MSGD0U,MSGDBT,MSGND
0014            IERR=0
0015            WRITE (LU2'IXDIR,ERR=100) IDIR
0016         500 CONTINUE
0017            RETURN
0018         100 CONTINUE
0019            WRITE (7,2000) MSG1,MSGW,MSGDI,MSGND,ID
0020        2000 FORMAT (1X,41A1,I6)
0021            IERR=1
0022            GO TO 500
0023            END
```

2.14 SUBROUTINE WRDIR

POSITION AND WRITE DIRECT ACCESS DIRECTORY FILE FOR SEGMENT



ORIGINAL PAGE IS
OF POOR QUALITY

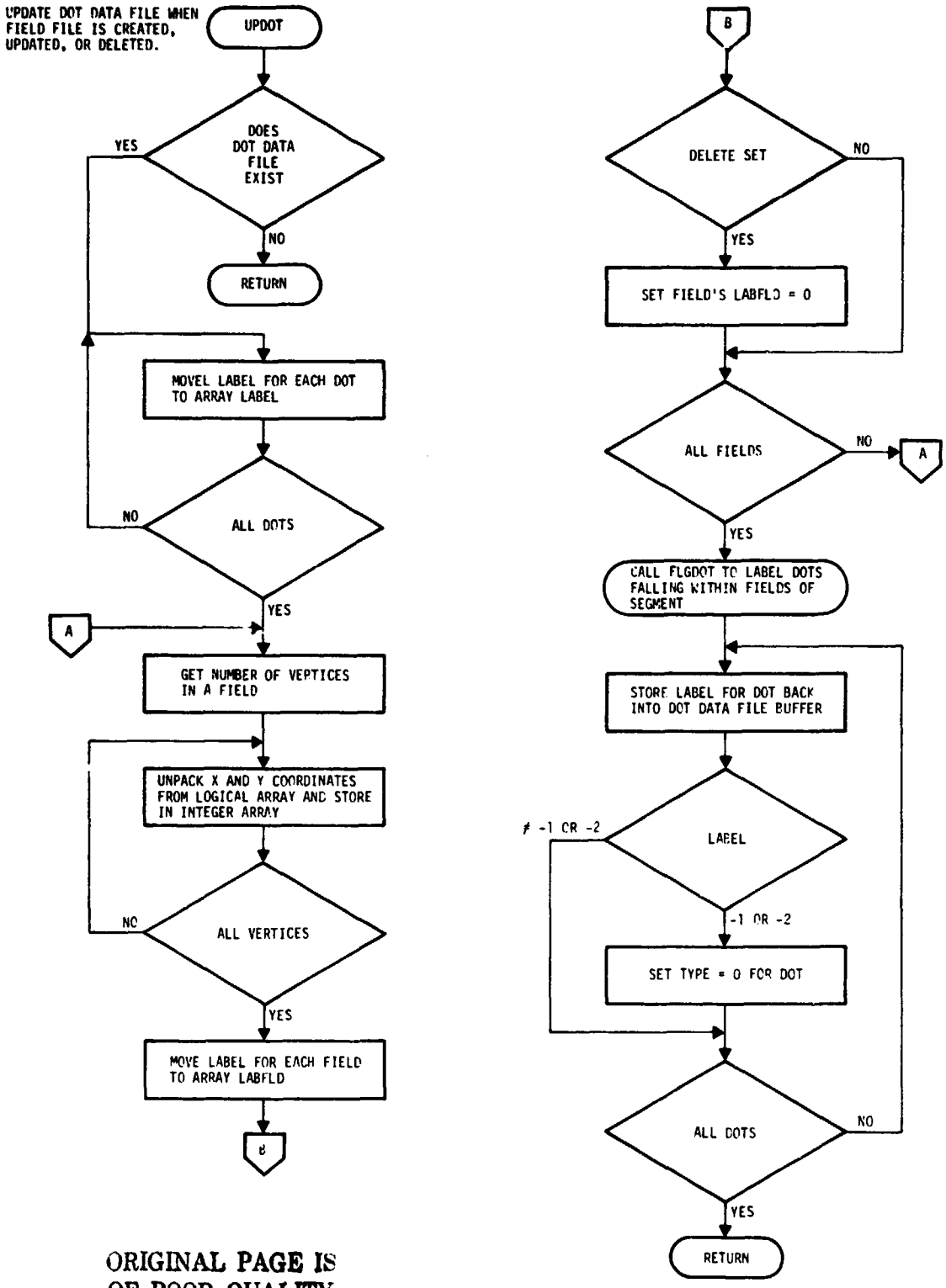
2.15 SUBROUTINE UPDOT

```
FØRTRAN IV+PLUS V02-04          08:50:09    11-MAY-77          PAGE 77
FLDUPD.FTN      /TR:BL0CKS/WR
0001            SUBR0UTINE UPD0T (I0PT)
                CCC
                C
                C         UPDATE D0T DATA FILE WHEN D0/DL FIELD IS DELETED
                C
                CCC
                C         I0PT = 1 SEGMENT DELETED
                C         I0PT = 2 N0RMAL ENTRY
0002            INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0003 *          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     *         1,MAXV=11,ND0TS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *         2N0SPWD=6,N0DTWD=10
0004            PARAMETER NBD0T=10868,NBPD0T=52
0005            C0MM0N /C0MD0T/ IXD0T,LD0T(NBD0T)
0006            BYTE LD0T
0007            PARAMETER NBDR=512,NBTBL=400,NBD0U=1450
0008            PARAMETER NBD0UE=30
0009            C0MM0N /BUFRS/ IXDIR,IBDIR,IXINX,IXD0U,
                1  LDTBL(NBTBL),LDIR(NBDR),LD0DU(NBD0U)
0010            BYTE LDIR,LD0DU
0011            C0MM0N /FLDC0M/ IVRTC(2,MAXV)
0012              C0MM0N /C0M3/ NFLD,NC00RD
0013            DIMENSI0N LABEL (ND0TS)
0014              DIMENSI0N IVERTX(2,MAXV,MAXFLD),LABFLD(MAXFLD)
0015              DIMENSI0N NVF(MAXFLD)
0016              BYTE L1
0017            BYTE LZER0
0018            DATA LZER0 /0/
                C         D0ES D0T DATA FILE EXIST F0R SEGMENT
0019            IF (IXD0T .EQ. 0) G0 T0 500
0020          200 C0NTINUE
                C         CHECK AND SET D0TS F0R FIELDS
0021            K=3
0022            D0 205 I=1,ND0TS
0023            LABEL(I)=LD0T(K)
0024            K=K+NBPD0T
0025          205 C0NTINUE
0026              K=9
0027              D0 604 I=1,NFLD
0028              NVT=LD0DU(K)
0029              NVF(I)=NVT
0030              L=1
0031              D0 602 J=1,NVT
0032              L1=LD0DU(K+L)
0033              IVERTX(1,J,I)=IBYTE(0,L1)
0034              L1=LD0DU(K+L+1)
0035              IVERTX(2,J,I)=IBYTE(0,L1)
0036              L=L+2
0037          602 C0NTINUE
0038              LABFLD(I)=LD0DU(K+1)
0039              IF (I0PT .EQ. 1) LABFLD(I)=0
0040              K=K+NBD0UE
0041          604 C0NTINUE
0042            CALL FLGD0T (NFLD,NVF,IVERTX,LABFLD,LABEL)
0043            K=3
0044            D0 214 I=1,ND0TS
```

```
FLDUPD.FTN      /TR:BLØCKS/WR
0045            LDØT(K)=LABEL(I)
0046            K=K+NBPDØT
0047        214 CØNTINUE
0048        500 CØNTINUE
0049            RETURN
0050            END
```

ORIGINAL PAGE IS
OF POOR QUALITY

2.15 SUBROUTINE UPDOT



ORIGINAL PAGE IS
OF POOR QUALITY

2.16 SUBROUTINE JULIAN

```
FØRTRAN IV+PLUS V02-04          08:49:34   11-MAY-77          PAGE 67
FLDUPD.FTN      /TR:BL0CKS/WR
0001            SUBR0UTINE JULIAN (YR,M0,DY,JUL!0)
0002            !MPL!CIT INTEGER (A-Z)
0003            DIMENSI0N C0NTAB(12)
0004              DATA C0NTAB /0,31,60,91,121,152,182,213,
               1 244,274,305,335/
0005            JUL!0=C0NTAB(M0)+DY
0006            XX=YR+1900
0007            YY=XX-((XX/4)*4)
0008            IF (YY .3T. 0) JUL!0=JUL!0+1
0009            RETURN
0010            END
```

2.16 SUBROUTINE JULIAN

A flowchart is not furnished for this subroutine.

2.17 SUBROUTINE FLDINT

```
FØRTRAN IV+PLUS V02+04          08:49:49    11+MAY+77          PAGE 73
FLDUPD.FTN      /TRIBLØCKS/WR
0001            SUBRØUTINE FDLINT(FIELD,NPTS,FL,YLINE,NSAMP,JJ)
                C!
                C!    THIS SUBRØUTINE WILL RETURN THE PIXEL NUMBERS ØF THØSE
                C!    PIXELS ØN A A GIVEN LINE THAT ARE CØNTAINED WITHIN THE
                C!    BØUNDARIES ØF A  NØN+RECTANGULAR FIELD
                C!
                C!
                C!    INPUT    FIELD +  NØN+RECTANGULAR FIELD TABLE
                C!                      ALL THE VERTICES MUST BE IN CLØCKWISE
                C!                      ØRDER AND THE LAST VERTEX HAS TØ BE EQUAL
                C!                      TØ THE FIRST VERTEX FØR FIELD CLØSURE
                C!                      THE FIRST VERTEX MUST HAVE MINIMUM
                C!                      PIXEL VALUE
                C!             NPTS  + NØ ØF PØINTS ØF THE N+R FIELD
                C!             YLINE + SCAN LINE NUMBER
                C!
                C!    ØUTPUT   FL    + ARRAY CØNTAINING THE ØRDERED PIXEL INTERCEPTS
                C!             NSAMP + NØ ØF SAMPLES CØNTAINED IN THE FIELD ØF
                C!                      A GIVEN SCAN LINE
                C!             JJ    + THE LENGTH ØF THE ARRAY FL
                C!
0002            PARAMETER MAXI=8
0003            DIMENSIØN FIELD(2,NPTS),FL(MAXI)
0004            INTEGER X1,X2,Y1,Y2,XX,FL,FIELD,YLINE
0005            INTEGER XNM1,YNM1,XNP2,YNP2
0006            IF(NPTS.EQ.2)GØ TØ 35
                C+    ØNE VERTEX FIELD
0007            L= YLINE
0008            DØ 7 N = 1,MAXI
0009          7 FL(N) = 0
0010            NPTSE = NPTS+1
0011            I = 1
0012            JJ = 0
0013        100 X1=FIELD(1,I)
0014            Y1=FIELD(2,I)
0015            J = I+1
0016            X2=FIELD(1,J)
0017            Y2=FIELD(2,J)
0018            IF ( I .EQ. 1 ) GØ TØ 200
0019            IM1 = I+1
0020            XNM1=FIELD(1,IM1)
0021            YNM1=FIELD(2,IM1)
0022            GØ TØ 300
0023        200 XNM1=FIELD(1,NPTSE)
0024            YNM1=FIELD(2,NPTSE)
0025        300 IP1 = I+1
0026            XNP1=FIELD(1,IP1)
0027            YNP1=FIELD(2,IP1)
0028            IF ( I .EQ. NPTSE) GØ TØ 400
0029            IP2 = I+2
0030            XNP2=FIELD(1,IP2)
0031            YNP2=FIELD(2,IP2)
0032            GØ TØ 500
0033        400 XNP2=FIELD(1,2)
0034            YNP2=FIELD(2,2)
0035        500 IF ( Y1 .EQ. Y2 ) GØ TØ 1000
```

```
FLDUPD,FTN      /TRIBLØCKS/WR
0036            IF((L.EQ.Y2).AND.(Y2.EQ.YNP2)) GØ TØ 2000
0037            IF((L.EQ.Y1).AND.(Y1.EQ.YNM1)) GØ TØ 2000
0038            RL = L
0039              RX1 = X1
0040              RX2 = X2
0041              RY1 = Y1
0042              RY2 = Y2
0043              RXX =(((RL-RY1)*(RX2-RX1))/(RY2-RY1))+RX1
0044              XX = RXX+.5
0045            IF(Y1.LT.Y2)  GØ TØ 510
0046            XX=RXX
0047            IF((RXX-XX).GT..5) XX=XX+1
0048     510    CØNTINUE
0049            IF ((XX.GE. X1) .AND. (XX .LE. X2) ) GØ TØ 600
0050            IF ((XX.LE. X1) .AND. (XX .GE. X2) ) GØ TØ 600
0051     2000   I = I+1
0052            IF ( I .GT. NPTSE ) GØ TØ 5
0053            GØ TØ 100
0054     600    IF(L.LE.Y1.AND.L.GE.Y2) GØ TØ 700
0055            IF(L.LE.Y2.AND.L.GE.Y1) GØ TØ 700
0056            GØ TØ 2000
0057     700    JJ = JJ+1
0058            FL(JJ) = XX
0059            IF ( JJ .EQ. 1 ) GØ TØ 2000
0060            IF ( I .NE. NPTSE ) GØ TØ 3000
0061            IF(L.NE.Y2) GØ TØ 3000
0062            XNM1=X1
0063            YNM1=Y1
0064            X1=X2
0065            Y1=Y2
0066            X2=FIELD(1,2)
0067            Y2=FIELD(2,2)
0068            GØ TØ 3001
0069     3000   IF ( L .NE. Y1 ) GØ TØ 2000
0070     3001     IF ((Y1.LT. YNM1) .AND. (Y1 .GT. Y2 )) GØ TØ 4000
0071            IF ((Y1 .GT. YNM1) .AND. (Y1 .LT. Y2)) GØ TØ 4000
0072            GØ TØ 2000
0073     4000   FL(JJ) = 0
0074            JJ = JJ-1
0075            GØ TØ 2000
0076     1000   IF(L.NE.Y1) GØ TØ 2000
0077            IF(X1.GT.X2) GØ TØ 5000
0078            IF(YNM1.LT.Y1) GØ TØ 6000
0079            IF ( YNP2 .GT. Y2 ) GØ TØ 7000
0080            JJ = JJ+1
0081            FL(JJ) = X1
0082            GØ TØ 2000
0083     7000   JJ = JJ+1
0084            FL(JJ) = X1
0085            MM = JJ+1
0086            FL(MM) = X2
0087            JJ = MM
0088            GØ TØ 2000
0089     6000   IF ( YNP2 .LT. Y2 ) GØ TØ 2000
0090            JJ = JJ+1
0091            FL(JJ) = X2
```

```
FLDUPD,FTN      /TR:BLØCKS/WR
0092            GØ TØ 2000
0093      5000  IF ( YNM1 .LT. Y1 ) GØ TØ 9000
0094            IF ( YNP2 .GT. Y2 ) GØ TØ 2000
0095            JJ = JJ+1
0096            FL(JJ) = X2
0097            IF(NPTSE.EQ.2)FL(JJ)=X1
0098            GØ TØ 2000
0099      9000  IF ( YNP2 .GT. Y2 ) GØ TØ 8000
0100            JJ = JJ+1
0101            FL(JJ) = X1
0102            MM = JJ+1
0103            FL(MM) = X2
0104            JJ = MM
0105            GØ TØ 2000
0106      8000  JJ = JJ+1
0107            FL(JJ) = X1
0108            GØ TØ 2000
0109         5  NPTS1 = JJ-1
0110            DØ 29 NI = 1,NPTS1
0111            NP1 = NI+1
0112            DØ 29 NJ = NP1,JJ
0113            IF ( FL(NI) - FL(NJ) ) 29,29,28
0114        28  NTEMP = FL(NI)
0115            FL(NI) = FL(NJ)
0116            FL(NJ) = NTEMP
0117        29  CØNTINUE
0118            NSAMP = 0
0119            DØ 30 N = 1,JJ,2
0120            NN = N+1
0121            NSAMP = NSAMP+(FL(NN) -FL(N)+1)
0122        30  CØNTINUE
0123            RETURN
0124        35  IF(YLINE.NE.FIELD(2,1))RETURN
0125            FL(1)=FIELD(1,1)
0126            FL(2)=FIELD(1,1)
0127            NSAMP=1
0128            JJ=2
0129            RETURN
0130            END
```

2.17 SUBROUTINE FLDINT

See volume 3, section 11.2 for a description. A flowchart is not provided.

2.18 SUBROUTINE FLGDOT

```
FØRTRAN IV+PLUS V02+04          08:49:37    11+MAY+77          PAGE 69
FLDUPD.FTN       /TR:BLOCKS/WR
0001            SUBROUTINE FLGDØT (NØFLD,NV,VERTEX,FLDLAB,DLABEL)
          C       FLAGS DØTS WHICH LIE WITHIN D0/DU FIELDS
          C       WRITTEN BY RUTH MINTER
          C       NØFLD  - N0. ØF FIELDS
          C       NV     - N0. ØF VERTICES IN EACH FIELD
          C       VERTEX - SPATIAL (PIXEL,LINE) CØØRDINATE ØF EACH VERTEX
          C       FLDLAB - TWØ CHARACTER LABEL (D0 ØR DU) FØR EACH FIELD
          C       DLABEL - CATEGØRY INDICATØR FØR EACH DØT
0002            IMPLICIT INTEGER (A-Z)
0003              INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0004 +          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     +         1,MAXV=11,NDØTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     +         2NØSPWD=6,NUDTWD=10
0005              DIMENSIØN NV(MAXFLD), VERTEX(2,MAXV,MAXFLD),
                 1  FLDLAB(MAXFLD),DLABEL(NDØTS),FL(8),LB(MAXFLD),LE(MAXFLD)
0006              DIMENSIØN SB(MAXFLD),SE(MAXFLD)
          C        RESET ALL DØTS WITH D0 ØR DU LABEL
0007              D0 10 I=1,NDØTS
0008              IF (DLABEL(I) .EQ. -1) DLABEL(I)=0
0009              IF (DLABEL(I) .EQ. -2) DLABEL(I)=0
0010           10 CØNTINUE
0011              D0 15 I=1,MAXFLD
0012              LB(I)=1000
0013              LE(I)=0
0014              SB(I)=1000
0015              SE(I)=0
0016           15 CØNTINUE
0017              D0 20 J=1,NØFLD
0018              NVT=NV(J)
0019              D0 20 K=1,NVT
0020              LB(J)=MIN(LB(J),VERTEX(2,K,J))
0021              LE(J)=MAX(LE(J),VERTEX(2,K,J))
0022              SB(J)=MIN(SB(J),VERTEX(1,K,J))
0023              SE(J)=MA (SE(J),VERTEX(1,K,J))
0024           20 CØNTINUE
0025              NP=NPIX/DSSKIP
0026              D0 50 J=1,NØFLD
0027              DØT=0
0028              D0 40 LINE=10,NLIN,DLSKIP
0029              D0 30 IS=10,NPIX,DSSKIP
0030                DØT=(LINE/DLSKIP-1)*NP+IS/DSSKIP
0031                IF (LINE .LT. LB(J)) G0 T0 40
0032                IF (LINE .GT. LE(J)) G0 T0 40
0033              IF (IS .LT. SB(J)) G0 T0 30
0034              IF (IS .GT. SE(J)) G0 T0 30
0035              CALL FDLINT (VERTEX(1,1,J),NV(J),FL,
                 1  LINE,NSAMP,JJ)
0036                D0 30 I=1,JJ,2
0037                IF (IS .LT. FL(I)) G0 T0 30
0038                IF (IS .GT. FL(I+1)) G0 T0 30
0039                DLABEL(DØT)=FLDLAB(J)
0040          9000    FØRMAT (10X,'DLABEL ',2I4)
0041           30 CØNTINUE
0042           40 CØNTINUE
0043           50 CØNTINUE
0044              RETURN
```

```
FØRTRAN V/◦PLUS V02◦04          08149137    11◦MAY◦77          PAGE 70
FLDUPD,FTN       /TRIBLOCKS/WR
0045             END
```

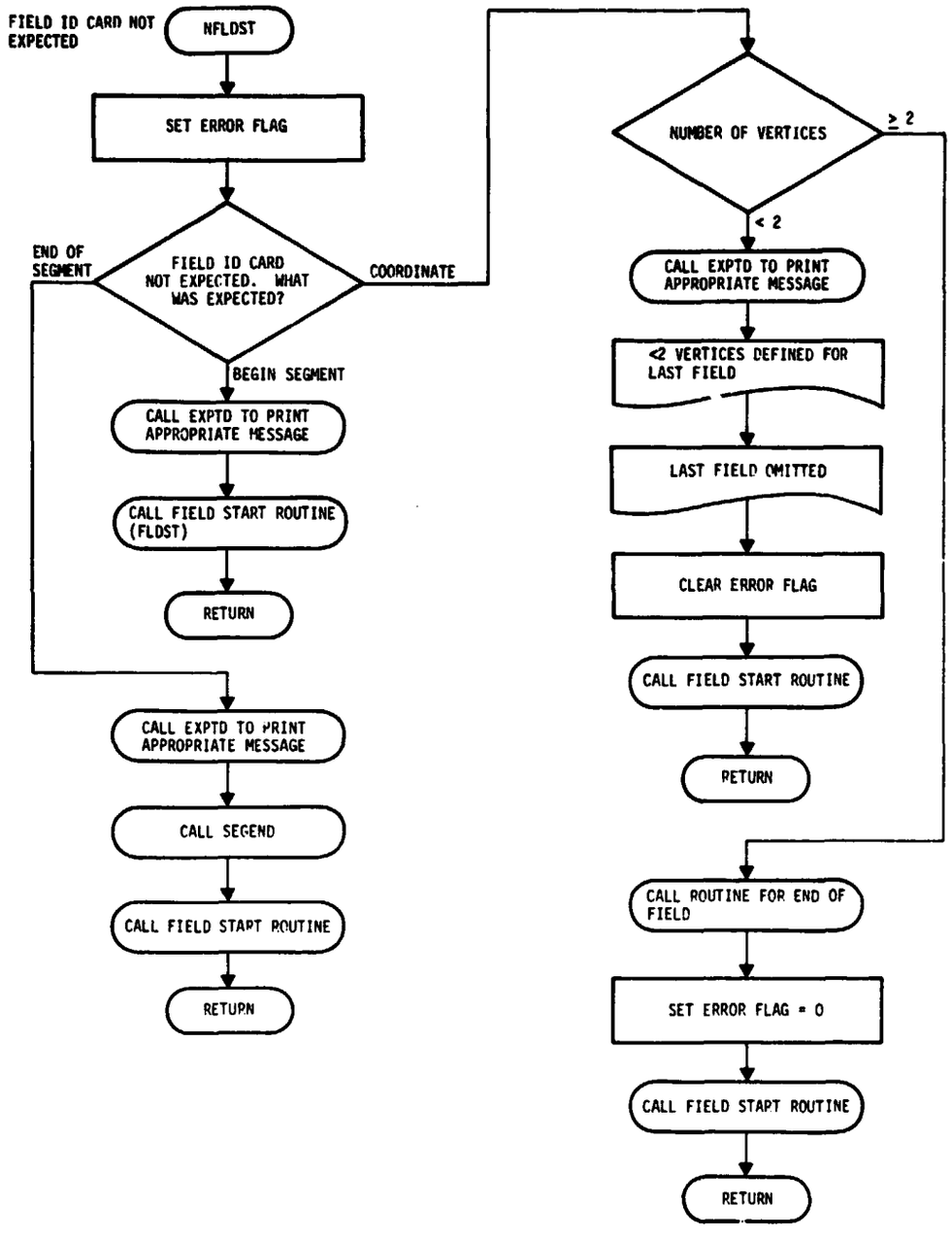
2.18 SUBROUTINE FLAGDOT

See volume 3, section 11.1 for a description. A flowchart is not provided.

2.19 SUBROUTINE NFLDST

```
FØRTRAN IV+PLUS V02+04          08:48:35    11-MAY-77          PAGE 40
FLDUPD.FTN      /TR:RL2CKS/WR
0001            SUBROUTINE NFLDST
                CCC
                C
                C       FIELD START (FLDSTART) WAS READ BUT WAS NOT EXPECTED
                C
                CCC
0002            COMMON /COM1/ IACTV,IACDI,NEWL,INDEXP(4),IERFLG,KCKBP,
               1 IDLTE,NDEXR,IDODU
0003            BYTE IDODU
0004            COMMON /COM3/ NFLD,NCORD
0005            COMMON /FILES/ IN,IOUT,LU1,LU2,LU3,LU4,MOUNT
                C       SET ERROR FLAG
0006            IERFLG=1
                C       WHAT WAS EXPECTED
0007            DO 10 I=1,4
0008            IF (INDEXP(I) .NE. 0) GO TO 15
0009         10 CONTINUE
                C       NOTHING
0010            GO TO 500
0011         15 CONTINUE
0012            GO TO (20,500,30,40), I
0013         20 CONTINUE
0014            CALL EXPTD(1)
0015            CALL FLDST
0016            GO TO 500
0017         30 CONTINUE
0018            IF (NCORD .LT. 2) GO TO 32
                C       COMPLETE LAST FIELD
0019            CALL FLDEND
0020            IERFLG=0
0021            CALL FLDST
0022            GO TO 500
0023         32 CONTINUE
0024            CALL EXPTD(3)
0025            IF (NEWL .NE. 0) WRITE (7,5000)
0026            WRITE (7,1032)
0027         1032 FORMAT (1H+,100X,'<2 VERTICES DEFINED')
0028            NEWL=1
0029         36 CONTINUE
0030            WRITE (7,1036)
0031         1036 FORMAT (81X,'LAST FIELD OMITTED',/)
0032            IERFLG=0
0033            CALL FLDST
0034            GO TO 500
0035         40 CONTINUE
0036            CALL EXPTD(4)
0037         5000 FORMAT (1H )
0038            IF (NEWL .NE. 0) WRITE (7,5000)
0039            WRITE (7,1050)
0040         1050 FORMAT (1H+,96X,'SEGEND SIMULATED',/)
0041            NEWL=1
0042            CALL SEGEND
0043            CALL FLDST
0044          500 CONTINUE
0045            RETURN
0046            END
```

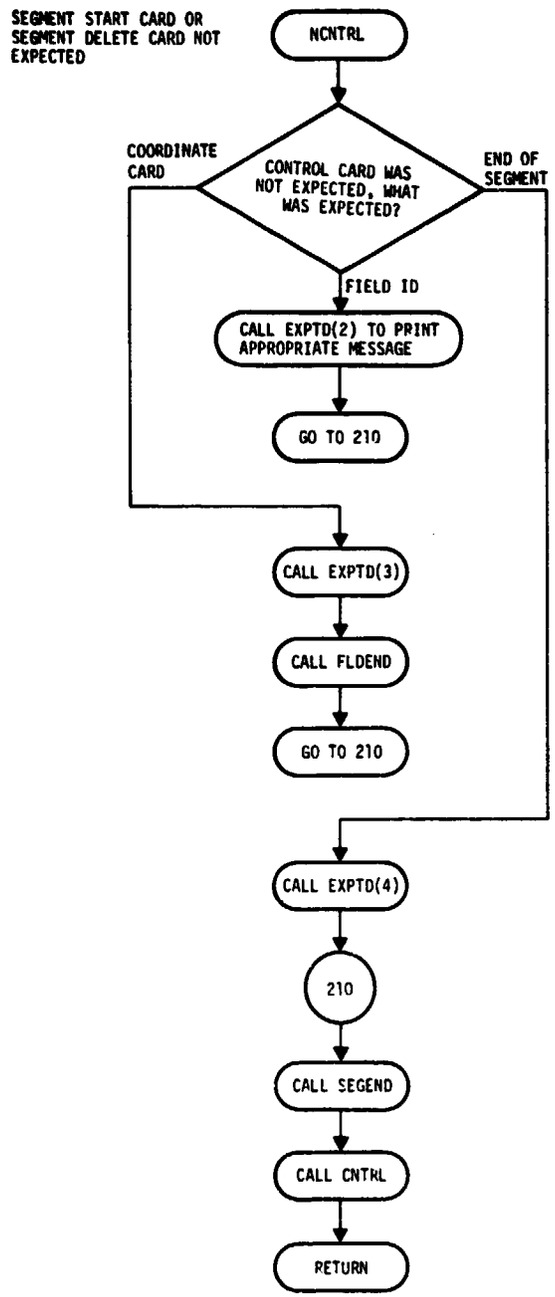
2.19 SUBROUTINE NFLDST



2.20 SUBROUTINE NCNTRL

```
FØRTRAN IV+PLUS V02-04          08148142    11-MAY-77          PAGE 42
FLDUPD.FTN      /TR:8L0CKS/WR
0001            SUBRØUTINE NCNTRL
        CCC
        C
        C         CØNTRØL CARD WAS READ BUT WAS NØT EXPECTED
        C
        CCC
0002            CØMMØN /CØM1/ IACTV,IACDI,NEWL,INDEXP(4),IERFLG,KCKBP,
               1 IDLTE,NDEXR,IDØDU
0003            BYTE IDØDU
0004            CØMMØN /FILES/ IN,IØUT,LU1,LU2,LU3,LU4,MØUNT
        C          WHAT WAS EXPECTED
0005            DØ 282 I=1,4
0006            IF (INDEXP(I) .NE. 0) GØ TØ 284
0007          282 CØNTINUE
0008            GØ TØ 500
0009          284 CØNTINUE
0010            GØ TØ (500,286,288,294), I
0011          286 CØNTINUE
0012              IF (INDEXP(3) .NE. 0) GØ TØ 288
0013            CALL EXPTD(2)
0014            GØ TØ 210
0015          288 CØNTINUE
0016            CALL EXPTD(3)
0017              CALL FLDEND
0018              GØ TØ 210
0019          294 CØNTINUE
0020            CALL EXPTD(4)
0021          210 CØNTINUE
0022            CALL SEGEND
0023            CALL CNTRL
0024          500 CØNTINUE
0025            RETURN
0026            END
```

2.20 SUBROUTINE NCNTRL



ORIGINAL PAGE IS
OF POOR QUALITY

2.21 SUBROUTINE NFIELD

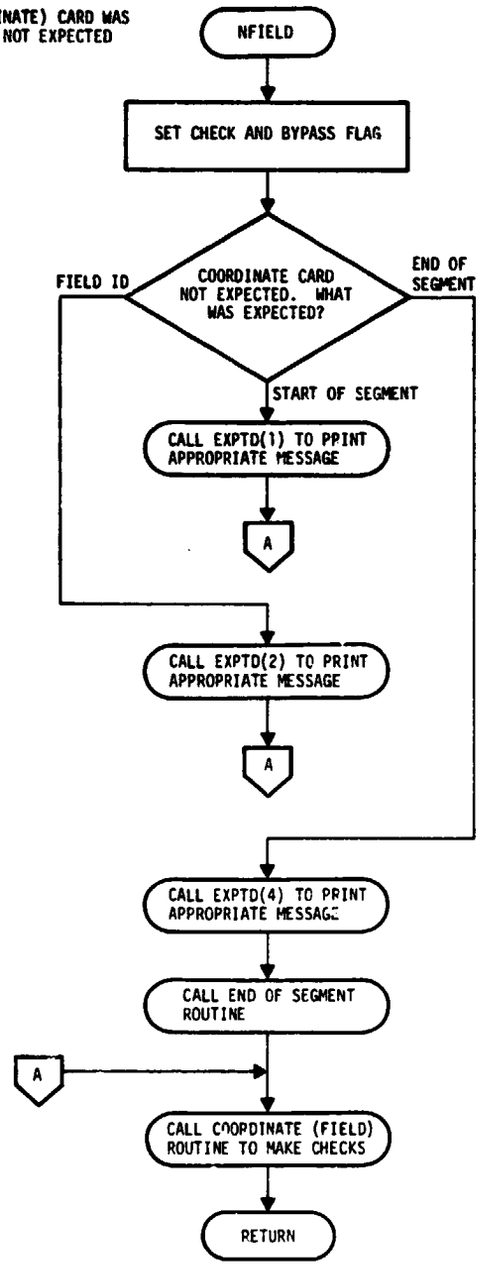
```
FØRTRAN IV+PLUS V02+04          08:48:45    11+MAY+77          PAGE 44
FLDUPD.FTN      /TR:BLØCKS/WR
0001            SUBR0UTINE \NFIELD\
        CCC
        C
        C         FIELD CARD WAS READ BUT WAS NØT EXPECTED
        C
        CCC
0002            CØMMØN /CØM1/ IACTV,IACDI,NEWL,INDEXP(4),IERFLG,KCKBP,
               1 IDLTE,NDEXR,IDØDU
0003            BYTE IDØDU
0004            CØMMØN /FILES/ IN,IØUT,LU1,LU2,LU3,LU4,MØUNT
        C          SET CHECK AND BYPASS FLAG
0005            KCKBP=1
        C          WHAT CARD WAS EXPECTED
0006             DØ 10 I=1,4
0007            IF (INDEXP(I) .NE. 0) GØ TØ 15
0008         10 CØNTINUE
        C          NØTHING
0009            GØ TØ 500
0010         15 CØNTINUE
0011            GØ TØ (20,30,500,40), I
0012         20 CØNTINUE
0013            CALL EXPTD(1)
0014            IF (NEWL .NE. 0) WRITE (7,5000)
0015       5000 FØRMAT (1H )
0016            WRITE (7,1021)
0017            NEWL=1
0018            GØ TØ 498
0019         30 CØNTINUE
0020            CALL EXPTD(2)
0021            IF (NEWL .NE. 0) WRITE (7,5000)
0022            WRITE (7,1021)
0023       1021 FØRMAT (1H+,101X,'FIELD ØUT ØF ØRDER')
0024            NEWL=1
0025            GØ TØ 498
0026         40 CØNTINUE
0027            CALL EXPTD(4)
0028            IF (NEWL .NE. 0) WRITE (7,5000)
0029            WRITE (7,1021)
0030            NEWL=1
0031            CALL SEGEND
0032        498 CØNTINUE
0033            CALL FIELD
0034        500 CØNTINUE
0035            RETURN
0036            END
```

FLDUPD.FTN /TRIBLØCKS/WR

```
0001          BLØCK DATA RWMSG
              CCC
              C
              C        READ/WRITE ERRØR MESSAGES
              C
              CCC
0002          CØMMØN /MSGRW/ MSG1(6),MSG2(12),MSGR(9),
             1  MSGW(9),MSGDI(9),MSGDØU(5),MSGDØT(8),MSGND(17)
0003           BYTE MSG1,MSG2,MSGR,MSGW,MSGDI,MSGDØU,MSGDØT,MSGND
0004           DATA MSG1 /1HE,1HR,1HR,1HØ,1HR,1H /
0005           DATA MSG2 /1HE,1HN,1HD,1H ,1HØ,1HF,1H ,1HF,1HI,
             1  1HL,1HE,1H /
0006           DATA MSGR /1H ,1HR,1HE,1HA,1HD,1HI,1HN,1HG,1H /
0007           DATA MSGW /1H ,1HW,1HR,1HI,1HT,1HI,1HN,1HG,1H /
0008           DATA MSGDI /1HD,1HI,1HR,1HE,1HC,1HT,1HØ,1HR,1HY/
0009           DATA MSGDØU /1HD,1HØ,1H/,1HD,1HU/
0010           DATA MSGDØT /1HD,1HØ,1HT,1H ,1HD,1HA,1HT,1HA/
0011           DATA MSGND /1H ,1HF,1HI,1HL,1HE,1H ,1HF,1HØ,1HR,
             1  1H ,1HS,1HE,1HG,1HM,1HE,1HN,1HT/
0012           END
```

2.21 SUBROUTINE NFIELD

FIELD (COORDINATE) CARD WAS READ BUT WAS NOT EXPECTED

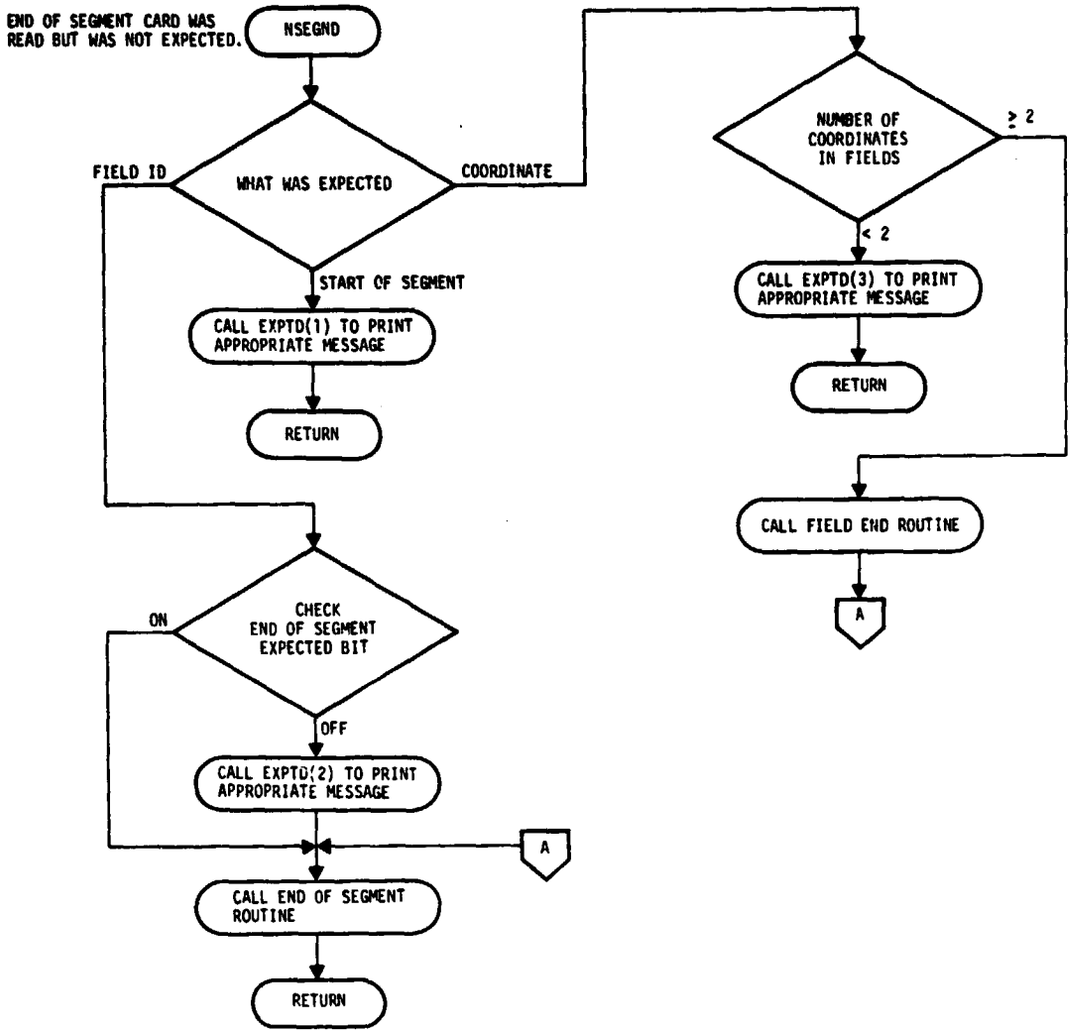


ORIGINAL PAGE IS OF POOR QUALITY

2.22 SUBROUTINE NSEGND

```
FØRTRAN IV+PLUS V02+04          08140150    11+MAY+77          PAGE 46
FLDUPD.FTN      /TR:PLØCKS/WR
0001            SUBRØUTINE NSEGND
                CCC
                C
                C       SEGEND CARD WAS READ BUT WAS NØT EXPECTED
                C
                CCC
0002                COMMØN /CØM1/ IACTV,IACDI,NEWL,INDEXP(4),IERFLG,KCKBP,
                   1 IDLTE,NDEXR,IDØDU
0003                COMMØN /CØM3/ NFLD,NCØØRD
0004                COMMØN /FILES/ IN,IØUT,LU1,LU2,LU3,LU4,MØUNT
0005                BYTE IDØDU
                C       WHAT WAS EXPECTED
0006                DØ 10 I=1,4
0007                IF (INDEXP(I) .NE. 0) GØ TØ 15
0008             10 CØNTINUE
0009                GØ TØ 500
0010             15 CØNTINUE
0011                GØ TØ (20,30,40,490), I
0012             20 CØNTINUE
0013                CALL EXPTD(1)
0014                GØ TØ 500
0015             30 CØNTINUE
0016                IF (INDEXP(4) .NE. 0) GØ TØ 490
0017                CALL EXPTD(2)
0018                GØ TØ 490
0019             40 CØNTINUE
0020                IF (NCØØRD .GE. 2) GØ TØ 488
0021                CALL EXPTD(3)
0022                GØ TØ 500
0023            488 CØNTINUE
0024                CALL FLDEND
0025                GØ TØ 490
0026             50 CØNTINUE
0027                CALL EXPTD(4)
0028            490 CØNTINUE
0029                CALL SEGEND
0030            500 CØNTINUE
0031                RETURN
0032                END
```

2.22 SUBROUTINE NSEGND



3. DOT DATA UPDATE PROGRAM DOTUPD

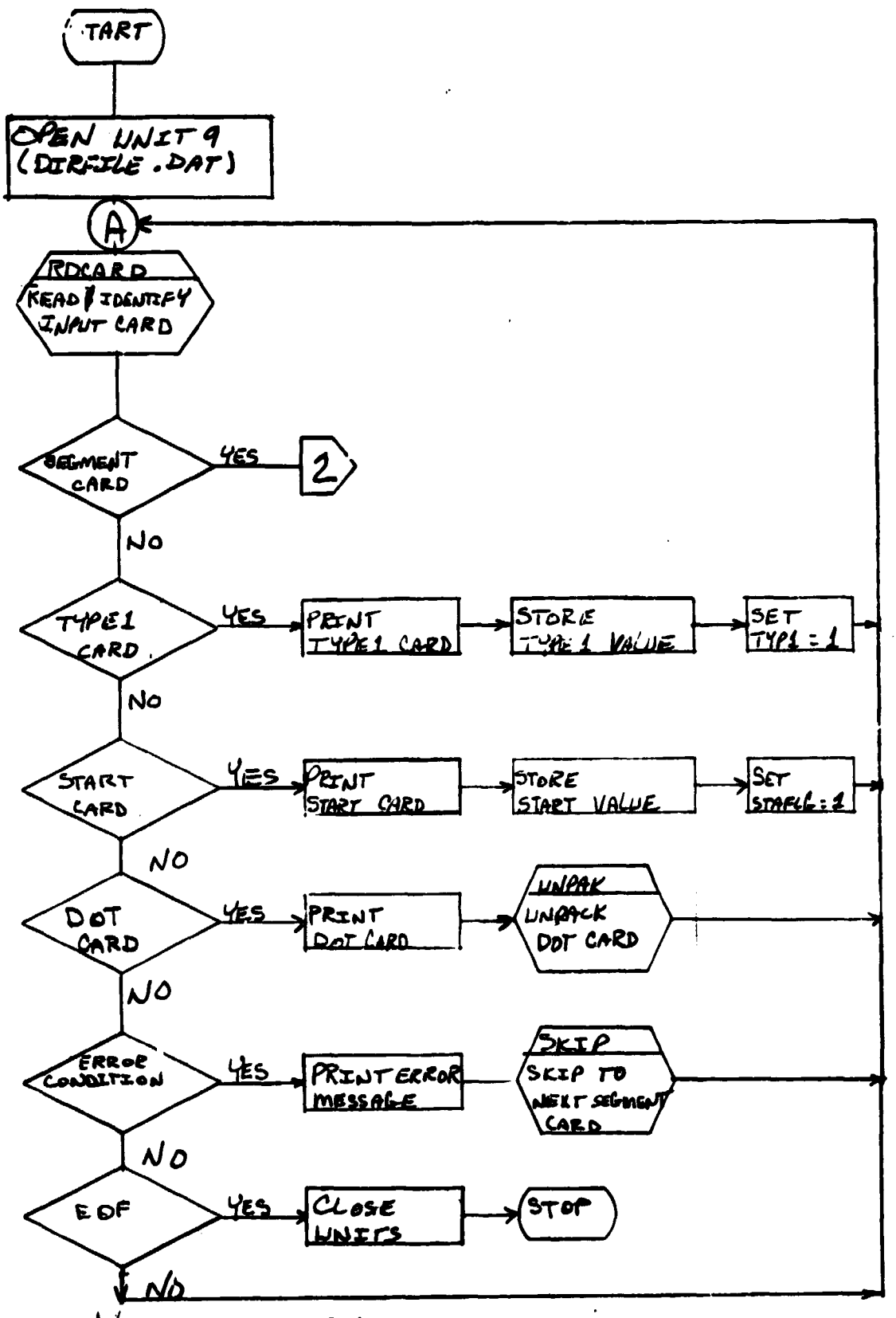
```
HFØRTRAN IV=PLUS V02=04                    10114134    01-AUG-77          PAGE 1
DØTUPD,FTN       /TRIBLØCKS/WR
0001             PRØGRAM DØTUPD
0002             IMPLICIT INTEGER(A-Z)
0003             INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0004 *          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFL=50
     *         1,MAXV=11,NDØTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *         2NØSPWD=6,NØDTWD=10
0005             CØMMØN/INFØ1/DØTBUF(3,NDØTS),KNT
0006             DIMENSIØN CARD(40),FILENM(15)
0007             DATA SEGØLD/-9/
0008             DATA SEGFLG/0/,DØTFLG/0/,TYPFLG/0/,STAFLG/0/
0009             DATA FILENM/2HDB,2H2:,2HF3,2HØC,2H,3,2HØØ,
                 *  2H] ,2H  ,2H D,2HD0,2HTS,2H.D,2HAT,24)1,0/
                 C
0010             CALL ASSIGN(3,'F#R001,DAT')
                 C          ØPEN THE DIRFILE,DAT FILE
0011             ØPEN(UNIT=9,NAME='[300,300]DIRFILE,DAT)1',TYPE='ØLD',
                 *  ACCESS='DIRECT',MAXREC=200,RECØRDSIZE=128)
                 C
                 C
                 C
                 C  READ AND IDENTIFY A CARD
                 C
0012     100     IDENT=RDCARD(CARD)
0013             GØ TØ (200,300,400,500,1250,600),IDENT
0014     200     CØNTINUE
                 C
                 C  THIS SECTIØN HANDLES SEGMENT CARDS
                 C
0015             SEGFLG=SEGFLG+1
0016             IF(SEGFLG .EQ. 2) GØ TØ 700
                 C
                 C  IF SEGFLG=1 PRØCESS SEGMENT CARD NOW
                 C  IF SEGFLG=2 PRØCESS SEGMENT CARD ON NEXT ITERATION
                 C
0017             PRINT 202
0018     202     FØRMAT(1HØ)
0019             PRINT 203,CARD
0020     203     FØRMAT(5X,40A2)
0021             DECØDE(15,205,CARD,ERR=1250)SEGNUM,INIT
0022     205     FØRMAT(9X,I4,1X,A1)
0023             SEGFLG=1
0024             IF(SEGØLD .EQ. SEGNUM)GØ TØ 100
0025             SEGØLD=SEGNUM
0026             CALL SUBSTR(CARD,10,4,FILENM,14,4)
0027             CALL DSKCHK(SEGNUM,PTR,DSKID,FLAG)
0028             IF(FLAG .EQ. 2) GØ TØ 1300
0029             IF(FLAG .EQ. 1) CALL ØPMESS(DSKID)
                 C  READ THE PRØPER DIRECTØRY RECØRD
0030             READ(9'PTR) DIRFIL
                 C
0031             CALL DIRLØD(PTR)
0032             GØ TØ 100
                 C
0033     300     CØNTINUE
                 C  THIS SECTIØN HANDLES TYPE1 CARDS
0034             PRINT203,CARD
```

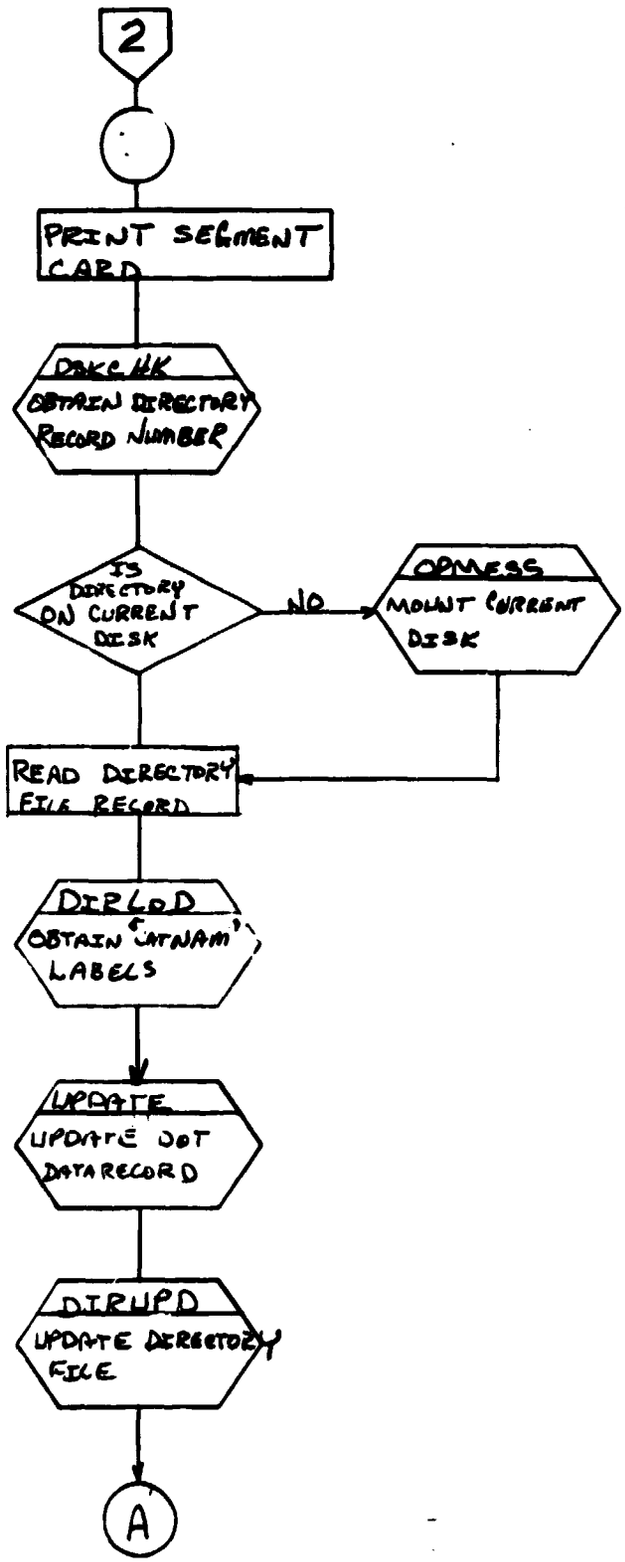
```
        C
0035            IF(SEGFLG .EQ. 0) G0 T0 1250
0036            IF(TYPFLG) 302,305,302
0037    302     PRINT 304
0038    304     F0RMAT(1H0,'ERR0R---0NLY 0NE TYPE1 VALUE ALL0WED',
     *          ' PER SEGMENT!)
0039            G0 T0 1270
0040    305     DEC0DE(10,310,CARD,ERR=1250) TYP1
0041    310     F0RMAT(6X,I2)
0042            TYPFLG=1
0043            G0 T0 100
        C
0044    400     C0NTINUE
        C  THIS SECTI0N HANDLES START CARDS
0045            PRINT 203,CARD
        C
0046            IF(SEGFLG .EQ. 0) G0 T0 1250
0047            IF(STAFLG) 410,420,410
0048    410     PRINT 415
0049    415     F0RMAT(1H0,'ERR0R---0NLY 0NE START VALUE ALL0WED',
     *          ' PER SEGMENT')
0050            G0 T0 1270
0051    420     DEC0DE(10,310,CARD,ERR=1250) START
0052            IF(START .GT. 0 .AND. START .LE. 60) G0 T0 100
0053            PRINT 425
0054    425     F0RMAT(1H0,9X,'ERR0R 0N START CARD---VALUE LESS THAN 1',
     *          ' 0R GREATER THAN 60')
0055            G0 T0 1270
        C
0056    500     C0NTINUE
        C  THIS SECTI0N HANDLES D0T CARDS
0057            PRINT 203,CARD
        C
0058            IF(SEGFLG .EQ. 0) G0 T0 1250
0059            IF(UNPAK(CARD,D0TFLG)) 1270,505,1270
0060    505     D0TFLG=1
0061            G0 T0 100
        C
0062    600     C0NTINUE
        C  THIS SECTI0N HANDLES END 0F FILE
        C
0063            IF(SEGFLG .EQ. 0 .0R. D0TFLG .EQ.0) STOP 1
0064            E0FFLG=1
        C
0065    700     C0NTINUE
        C  THIS SECTI0N UPDATES THE XXXX.D0TS.DAT FILE!ALS0 THE DIRECT0RY FILE
        C
0066            0PEN(UNIT=8,NAME=FILENM,TYPE='0LD',ACCESS='DIRECT',
     *          F0RM='UNF0RMATTED',RECORDSIZE=13,DISP0SE='SAVE',
     *          BL0CKSIZE=ND0TS,ASS0CIATEVARIABLE=V1)
0067            CALL UPDATE(TYP1,INIT)
0068            CALL DIRUPD(START,TYP1)
0069            CL0SE(UNIT=8,DISP0SE='SAVE')
0070            IF(E0FFLG .EQ. 1) STOP 2
        C
        C  INITIALIZE FLAGS
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
        C
0071    750     DØTFLG=0
0072            SEGFLG=0
0073            TYRFLG=0
0074            STAFLG=0
0075            GØ TØ 200
        C
        C  THIS SECTIØN FØR ERRØR MESSAGE AND RECØVERY
        C
0076    1250    PRINT 1260
0077    1260    FØRMAT(1HØ,9X,'IMPRØPER SEQUENCE,ØR,IMPRØPER DATA,ØR, ',
     *                      'CØLUMN MISALIGNMENT ØN THE FØLLØWING CARD')
0078            PRINT 203,CARD
0079    1270    PRINT 1275,SEGNUM
0080    1275    FØRMAT(1HØ,'PROCESSING ØF SEGMENT',I5,' WILL BE DISCØNTINUED ',
     *                  '--- PROCESSING WILL CONTINUE WITH A NEW ',
     *                  'SEGMENT ØR HALT WITH AN EØF')
0081            IF(SEGFLG .EQ. 2) GØ TØ 750
0082            IDENT=SKIP(CARD)
0083            GØ TØ(750,1250,1280),IDENT
0084    1280    PRINT 1285
0085    1285    FØRMAT(1HØ,'READ EØF AND TERMINATED')
0086            STØP 3
        C
        C       THIS SECTIØN HANDLES DISK CHECK FLAGS 1&2
        C       1= WRØNG DISK PACK MØUNTED
        C       2= SEGMENT NUMBER IS NØT IN THE SYSTEM AT ALL
        C            WHICH IS A WIPE ØUT!
        C
0087    1300    TYPE 1310,SEGNUM
0088    1310    FØRMAT(1HØ,'ERRØR---ILLEGAL SEGMENT ',I4,/,1X,'RUN ARØRTED')
0089            STØP
0090            END
```

3. DOT DATA UPDATE PROGRAM DOTUPD




2
PRINT SEGMENT CARD
DSKCHK
OBTAIN DIRECTORY RECORD NUMBER
IS DIRECTORY ON CURRENT DISK
NO
OPMESS
MOUNT CURRENT DISK
READ DIRECTORY FILE RECORD
DIRLOD
OBTAIN 'CATNAM' LABELS
UPDATE
UPDATE DOT DATA RECORD
DIRUPD
UPDATE DIRECTORY FILE
A

3.1 FUNCTION UNPAK

```
FORTRAN IV+PLUS V02-04          10114147     01-AUG-77          PAGE 5
D0TUPD.FTN      /TR:BLOCKS/WR
0001            INTEGER FUNCTION UNPAK(CARD,D0TFLG)
0002            IMPLICIT INTEGER(A-Z)
0003            INCLUDE'SY:[300,3]CAMSPARAM.INC'
0004 *         PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     *        1,MAXV=11,ND0TS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *        2N0SPHD=4,N0DTHD=10
0005            COMMON/INF01/D0TBUF(3,ND0TS),KNT
0006            LOGICAL*1 LABL0G(2),V1,V2,SPACE
0007            DIMENSION CARD(40),D0TS(72),LABEL(1)
0008            EQUIVALENCE(LABEL,LABL0G)
0009            DATA D0TS(72)/1H /,SPACE/1H /
0010            DECODE(80,10,CARD,ERR=200)TYPE,LABEL,(D0TS(I),I=1,71)
0011    10      FORMAT(4X,I1,1X,A2,1X,71A1)
0012            IF(LABL0G(1) .NE. SPACE)GO TO 14
0013            LABL0G(1)=LABL0G(2)
0014            LABL0G(2)=SPACE
0015    14      IF(D0TFLG .EQ. 1) GO TO 15
0016            KNT=0
0017    15      CONTINUE
        C
        C
0018            IF(TYPE .LT. 0 .OR. TYPE .GT.2) GO TO 200
0019            K=0
0020            D0TNUM=0
0021            DO 80 J=2,72
0022            V1=D0TS(J-1)
0023            V2=D0TS(J)
0024            IF(V1 .EQ. SPACE) GO TO 80
0025            K=K+1
0026            IF(K .GT. 3) GO TO 200
0027            DECODE(1,50,V1,ERR=200)N1
0028    50      FORMAT(I1)
0029            D0TNUM=D0TNUM*10 + N1
0030            IF(V2.SPACE) 80,70,80
0031    70      IF(D0TNUM .LT. 1 .OR. D0TNUM .GT. ND0TS)GO TO 200
0032            KNT=KNT+1
0033            D0TBUF(1,KNT)=D0TNUM
0034            D0TBUF(2,KNT)=TYPE
0035            IF(CATL0G(LABEL,LABNUM) .EQ. 1) GO TO 215
0036            D0TBUF(3,KNT)=LABNUM
0037            D0TNUM=0
0038            K=0
0039    80      CONTINUE
        C
0040            UNPAK=0
0041            RETURN
0042    200     PRINT 210,CARD
0043    210     FORMAT(10X,'ERROR(S) ON THE FOLLOWING CARD,PLEASE CHECK ALL ',
              *          'LETTERS AND NUMBERS FOR PROPER FIELD PLACEMENT',/,
              *       10X,40A2)
0044            GO TO 215
0045    215     UNPAK=1
0046            RETURN
0047            END
```

ORIGINAL PAGE IS
OF POOR QUALITY

3.1 FUNCTION UNPACK

A flow chart for this function is not available.

3.2 FUNCTION CATLOG

```
FØRTRAN IV+PLUS V02-04          10:14:55    01-AUG-77          PAGE 7
DBTUPD.FTN      /TR:BL@CKS/WR
0001            INTEGER FUNCTIØN CATL@G(LABEL,LABNUM)
0002            C@MM@N /CATNAM/ CATNAM(61)
0003            INTEGER CATNAM
0004            IF(LABEL .NE. BLANK) G@ T@ 50
0005            J=0
0006            G@ T@ 80
0007    50      D@ 100 J=1,60
0008            IF(CATNAM(J) .LE. 0) G@ T@ 200
0009            IF(LABEL .NE. CATNAM(J)) G@ T@ 100
0010    80      LABNUM=J
0011            CATL@G=0
0012            RETURN
0013    100     C@NTINUE
0014            PRINT 110,LABEL
0015    110     F@RMAT(1H0,'THE LABEL ',A2,' CAN N@T BE F@UND IN THE CATEG@RY ',
              * 'NAME TABLE.  THIS TABLE IS FULL',/,1X,
              * 'CAN N@T ACC@M@DATE ANY M@RE LABELS-----RUN AB@RTED')
0016            CATL@G=1
0017            RETURN
0018    200     PRINT 210,LABEL,J
0019    210     F@RMAT(1H0,'CREATING NEW CATEG@RY NAME',A3,/,1X,
              * 'P@SITI@N @F NEW CATEG@RY NAME=',I2)
0020            CATNAM(J)=LABEL
0021            LABNUM=J
0022            CATL@G=0
0023            RETURN
0024            END
```

3.2 SUBROUTINE CATLOG

ORIGINAL PAGE IS
OF POOR QUALITY

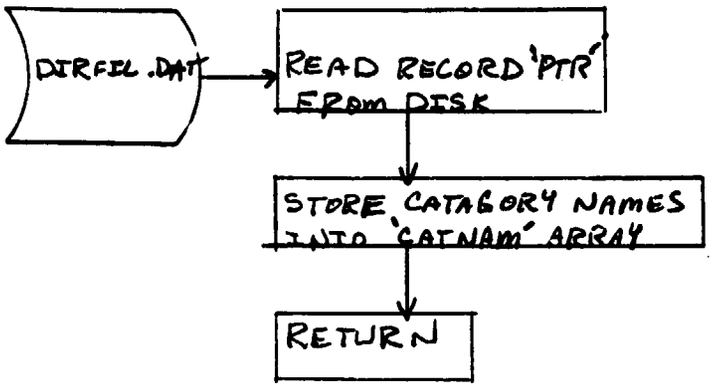
A flow chart for this subroutine is not available.

3.3 SUBROUTINE DIRLOD

```
FØRTRAN IV+PLUS V02-04          10114159     01-AUG-77          PAGE 9
DØTUPD.FTN      /TR:BLØCKS/KR
0001            SUBRØUTINE DIRLØD(PTR)
0002            IMPLICIT INTEGER(A-Z)
0003            CØMMØN /CATNAM/ CATNAM(61)
0004            DIMENSIØN DIRWD(256)
0005            LØGICAL*1 DIRLØG(512)
0006            EQUIVALENCE (DIRWD,DIRLØG)
0007            REC=PTR
0008            READ(9'REC) DIRWD
0009            K=0
0010            DØ 40 I=52,112
0011            K=K+1
0012            CATNAM(K)=DIRWD(I)
0013    40      CØNTINUE
0014            RETURN
        C
        C
0015            ENTRY DIRUPD(START,TYPE1)
0016            CALL IDATE(MØ,DAY,YR)
0017            DIRWD(36)=YR
0018            DIRWD(37)=JULIAN(MØ,DAY,YR)
0019            DIRLØG(75)=START
0020            DIRLØG(76)=TYPE1
0021            NØCAT=0
0022            K=0
0023            DØ 80 I=52,112
0024            K=K+1
0025            DIRWD(I)=CATNAM(K)
0026            IF(CATNAM(K) .NE. 0) NØCAT=NØCAT+1
0027    80      CØNTINUE
0028            DIRWD(51)=NØCAT
0029            WRITE(9'REC) DIRWD
0030            RETURN
0031            END
```

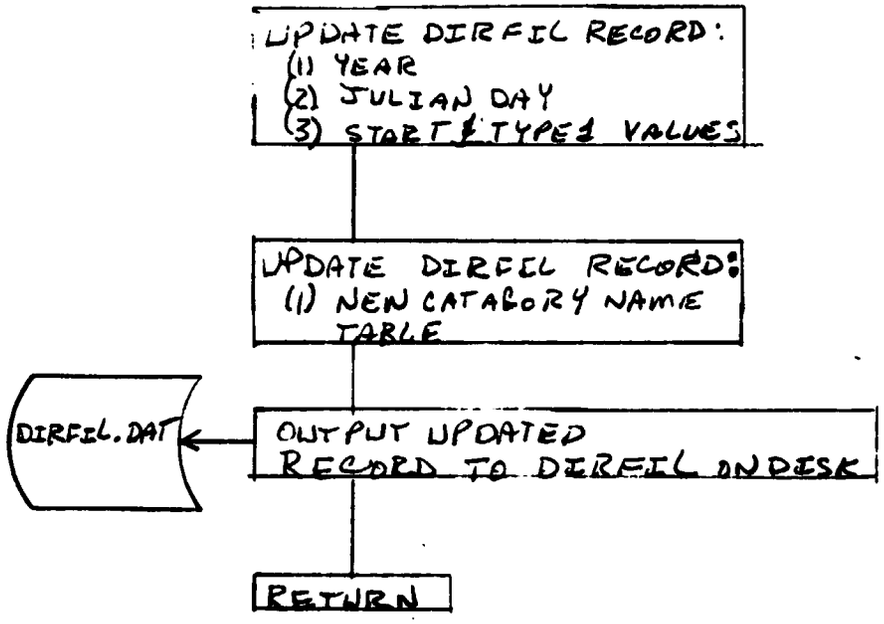
3.3 SUBROUTINE DIRLOD

PTR - POINTER TO PROPER RECORD IN DIRECTORY FILE.



ENTRY DIRUPD (START, TYPE1)

START - VALUES INPUT FROM DATA CARDS
TYPE1



3.4 FUNCTION JULIAN

```
FØRTRAN IV+PLUS V02-04          10:15:04     01-AUG-77          PAGE 11
DØTURD.FTN      /TR:BLØCKS/WR
0001            FUNCTIØN JULIAN(MØ,DAY,YR)
0002            INTEGER TØTAL,DAY,YR
0003            DIMENSIØN MDAYS(12)
0004            DATA MDAYS/31,0,31,30,31,30,31,31,30,31,30,31/
0005            MDAYS(2)=28
0006            IF(.NØT.(YR.EQ.0)) MDAYS(2)=29
0007            TØTAL=0
0008            LEN=MØ-1
0009            IF(LEN)10,10,20
0010    10      JULIAN=DAY
0011            RETURN
0012    20      DØ 30 I=1,LEN
0013            TØTAL=TØTAL+MDAYS(I)
0014    30      CØNTINUE
0015            JULIAN=TØTAL+DAY
0016            RETURN
0017            END
```

3.4 FUNCTION JULIAN

A flow chart for this function is not available.

3.5 DATA RANDOT

```
FØRTRAN IV+PLUS V02-04          10:15:06     01-AUG-77          PAGE 13
DØTUPD.FTN      /TR:BLØCKS/WR
0001              BLØCK DATA
0002              INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0003  *          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      *         1,MAXV=11,NDØTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *         2NØSPWD=6,NØDTWD=10
0004              CØMMØN /RANDØT/  RANDØT(NDØTS)
0005              INTEGER RANDØT
0006              DATA RANDØT/200,67,208,14,127,99,35,196,191,95,66,
                  *  137,122,186,114,61,80,206,30,90,172,199,
                  *  110,56,17,44,135,174,48,121,188,184,2,
                  *  112,132,4,181,43,60,158,63,20,198,94,
                  *  39,86,125,118,104,134,82,185,15,147,157,
                  *  77,75,161,3,76,187,202,8,136,73,89,
                  *  152,13,207,153,9,175,33,32,24,162,131,
                  *  42,18,49,203,36,1,155,96,87,193,171,
                  *  50,19,70,120,164,68,54,194,81,93,166,
                  *  173,22,105,180,115,5,27,46,12,38,16,
                  *  84,78,28,204,124,69,176,106,144,143,6,
                  *  141,72,59,10,201,71,190,107,150,40,88,
                  *  146,102,31,182,137,62,177,138,167,163,51,
                  *  34,129,151,186,113,119,23,52,21,109,150,
                  *  195,29,145,55,209,108,92,117,64,97,79,
                  *  91,45,128,165,26,116,53,139,198,111,160,
                  *  7,47,11,148,85,154,58,74,192,169,123,
                  *  98,178,140,205,101,65,103,57,126,179,142,
                  *  83,156,25,189,133,170,100,41,149,159,37/
0007              END
```

3.5 DATA RANDOT

A flow chart is not available.

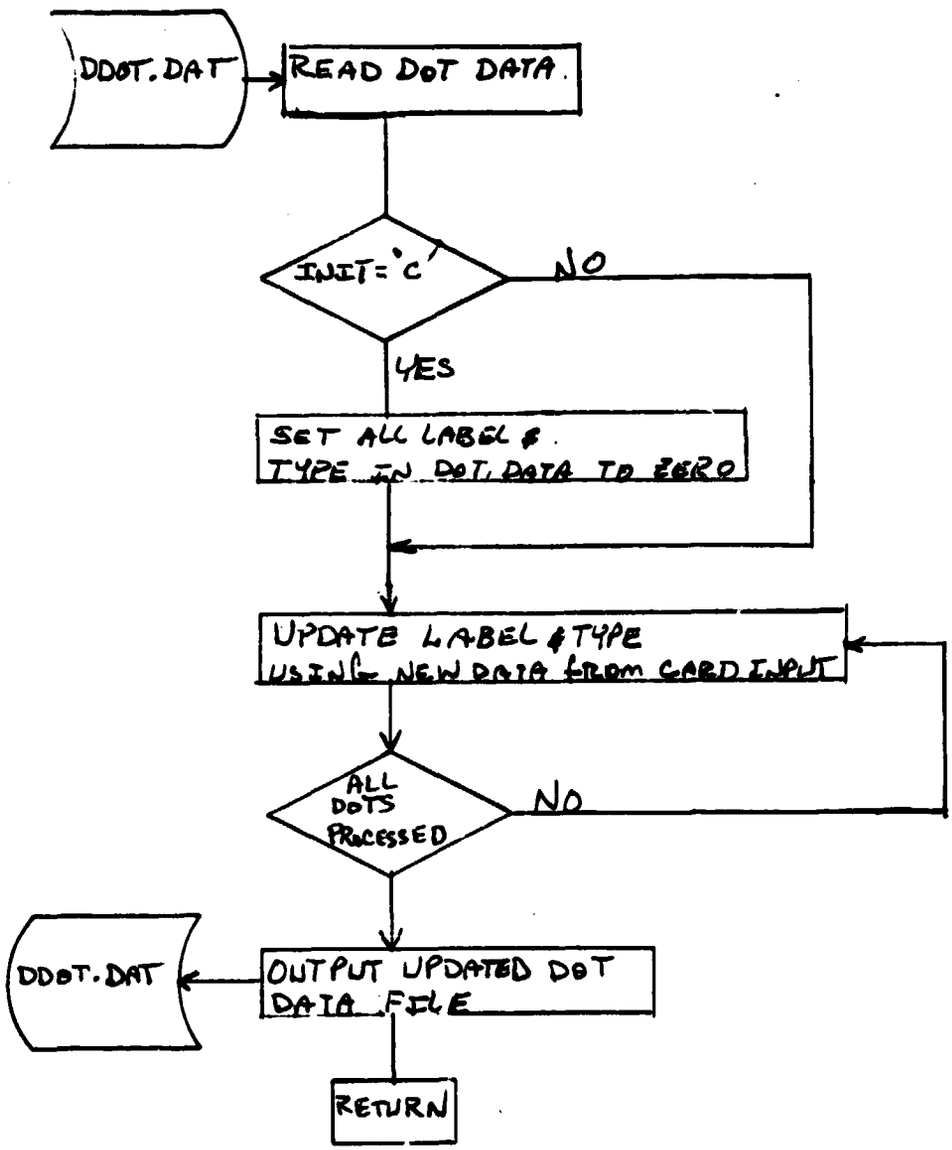
3.6 SUBROUTINE UPDATE

```
FØRTRAN IV+PLUS V02+04          10:15:10     01-AUG-77          PAGE 15
DØTUPD.FTN      /TR:BL0CKS/WR
0001            SUBR0UTINE UPDATE(TYP1,INIT)
0002            IMPLICIT INTEGER(A-Z)
0003            INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0004 *        PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     *       1,MAXV=11,ND0TS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAYACC=4,
     *       2N0SPWD=4,N0DTWD=10
0005            L0GICAL+1 DATA(52),W0RK(2,ND0TS)
        C
        C
0006            D0 705 J=1,ND0TS
0007            READ(8'J) DATA
0008            W0RK(1,J)=DATA(3)     !TYPE
0009            W0RK(2,J)=DATA(4)
0010    705     C0NTINUE
        C
0011            CALL ALLUPD(W0RK,TYP1,INIT)
        C
0012            D0 740 J=1,ND0TS
0013            READ(8'J) DATA
0014            DATA(3)=W0RK(1,J)
0015            DATA(4)=W0RK(2,J)
0016            WRITE(8'J) DATA
0017    740     C0NTINUE
0018            RETURN
0019            END
```

3.6 SUBROUTINE UPDATE

TYP1 - Same as TYPE1.

INIT - If equal to 'C', do complete update;
if equal to blank, do partial update.



3.7 SUBROUTINE ALLUPD

```
FORTRAN IV-PLUS V02-04          10:15:15    01-AUG-77          PAGE 17
D0TUPD.FTN      /TR:BLOCKS/WR
0001            SUBROUTINE ALLUPD(WORK,TYP1,INIT)
0002            IMPLICIT INTEGER(A-Z)
0003            INCLUDE 'SYS:[300,3]CAMSPARAM.INC'
0004 *        PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     *       1,MAXV=11,ND0TS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *       2N0SPND=6,N0DTHD=10
0005            COMMON /INF01/ D0TBUF(3,ND0TS),KNT
0006            COMMON /RAND0T/ RAND0T(ND0TS)
0007            LOGICAL*1 WORK(2,ND0TS)
      C
      C
0008            IF(INIT .NE. 'C' .AND. TYP1 .EQ. 0)G0 T0 30
0009            D0 20 J=1,ND0TS
0010            0LDLAB=W0RK(2,J)
0011            W0RK(1,J)=0
0012            IF(0LDLAB .EQ. -1 .0R. 0LDLAB .EQ. -2)G0 T0 20
0013            IF(INIT .EQ. 'C') W0RK(2,J)=0
0014    20      C0NTINUE
0015    30      D0 40 J=1,KNT
0016            D0TNUM=D0TBUF(1,J)
0017            TYPE=D0TBUF(2,J)
0018            LABEL=D0TBUF(3,J)
0019            0LDLAB=W0RK(2,D0TNUM)
0020            IF(0LDLAB .EQ. -1 .0R. 0LDLAB .EQ. -2)G0 T0 40
0021            IF(INIT .EQ. 'C' .0R. TYP1 .GT. 0) G0 T0 35
0022            IF(TYPE .GT. 0 .AND. LABEL .GT. 0) G0 T0 35
0023            W0RK(1,D0TNUM)=0
0024            W0RK(2,D0TNUM)=0
0025            G0 T0 40
0026    35      W0RK(1,D0TNUM)=TYPE
0027            W0RK(2,D0TNUM)=LABEL
0028    40      C0NTINUE
0029            IF(TYP1 .EQ. 0) G0 T0 130
0030            NHITS=0
0031            D0 100 J=1,ND0TS
0032            PT=RAND0T(J)
0033            LABEL=W0RK(2,PT)
0034            IF(LABEL .LE. 0) G0 T0 100
0035            W0RK(1,PT)=1
0036            NHITS=NHITS+1
0037            IF(NHITS .EQ. TYP1) G0 T0 110
0038    100     C0NTINUE
0039            TYP1=NHITS
0040    110     D0 120 J=1,ND0TS
0041            LABEL=W0RK(2,J)
0042            IF(LABEL .LE. 0) G0 T0 120
0043            TYPE=W0RK(1,J)
0044            IF(TYPE .LE. 0)W0RK(1,J)=2
0045    120     C0NTINUE
0046            RETURN
0047    130     NHITS=0
0048            D0 200 J=1,ND0TS
0049            TYPE=W0RK(1,J)
0050            IF(TYPE .EQ. 1) NHITS=NHITS+1
0051    200     C0NTINUE
0052            TYP1=NHITS
```

```
DBTUPD.FTN      /TR:BLOCKS/WR
0053            RETURN
0054            END
```

3.7 SUBROUTINE ALLUPD

A flow chart for this subroutine is not available.

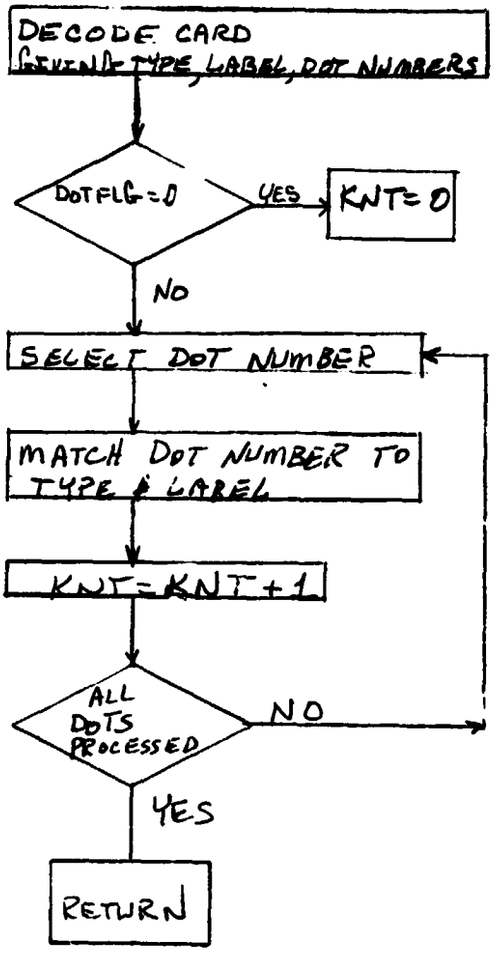
3.8 FUNCTION RDCARD

```
FØRTRAN IV-PLUS V02-04          10:15:22    01-AUG-77          PAGE 20
DØTUPD.FTN      /TR:BLØCKS/WR
0001            INTEGER FUNCTIØN RDCARD(CARD)
0002            IMPLICIT INTEGER(A-Z)
0003            DIMENSIØN CRDTYP(3,3),DØT(2),CARD(40)
0004            DATA CRDTYP/2HSE,2HGM,2HEN,
                *           2HTY,2HPE,2H1 ,
                *           2HST,2HAR,2HT /
0005            DATA DØT/2HD0,2HT /
0006    10      STAT=CARDIN(CARD)
0007            GØ TØ (20,100,110),STAT
0008    20      DØ 50 J=1,3
0009            IF(CØMPAR(CRDTYP(1,J),1,6,CARD,1)) 50,70,50
0010    50      CØNTINUE
0011            IF(CØMPAR(DØT,1,3,CARD,1))10,60,10
0012    60      J=4
0013    70      RDCARD=J
0014            RETURN
0015    100     RDCARD=5    !CARD READ ERRØR
0016            RETURN
0017    110     RDCARD=6    !END ØF FILE
0018            RETURN
0019            END
```

ORIGINAL PAGE IS
OF POOR QUALITY

3.8 FUNCTION RDCARD

CARD - An array to hold DOT CARD data.
DOTFLG - A flag to indicate first pass.



3.9 FUNCTION SKIP

```
FØRTRAN IV-PLUS V02-04          10:15:25     01-AUG-77          PAGE 22
DØTUPD.FTN                      /TR:BLØCKS/WR
0001                INTEGER FUNCTIØN SKIP(CARD)
0002                IMPLICIT INTEGER(A-Z)
0003                DIMENSIØN CRDTYP(3),CARD(40)
0004                DATA CRDTYP/2HSE,2HGM,2HEN/
0005     10         STAT=CARDIN(CARD)
0006                GØ TØ (20,40,50),STAT
0007     20         IF(CØMPAR(CRDTYP,1,6,CARD,1)) 10,30,10
0008     30         SKIP=1
0009                RETURN
0010     40         SKIP=2
0011                RETURN
0012     50         SKIP=3
0013                RETURN
0014                END
```

ORIGINAL PAGE IS
OF POOR QUALITY

3.9 FUNCTION SKID

A flow chart for this function is not available.

3.10 FUNCTION CARDIN

ORIGINAL PAGE IS
POOR QUALITY

```
FØRTRAN IV+PLUS V02-04          10:15:26     01-AUG-77          PAGE 24
DØTUPD.FTN      /TR:BLØCKS/WR
0001            INTEGER FUNCTIØN CARDIN(CARD)
0002            INTEGER CARD(40)
0003            READ(3,20,ERR=30,END=40) CARD
0004     20     FØRMAT(40A2)
0005            CARDIN=1
0006            RETURN
0007     30     PRINT 35,CARD
0008     35     FØRMAT(1HØ,9X,'CARD READER INDICATES THE FØLLØWING ',
               *                'CARD IN ERRØR',/,10X,80A1)
0009            CARDIN=2
0010            RETURN
0011     40     CARDIN=3
0012            RETURN
0013            END
```

3.10 FUNCTION CARDIN

A flow chart for this function is not available.

3.11 FUNCTION COMPAR

```
FØRTRAN IV-PLUS V02-04          10:15:28     01-AUG-77          PAGE 26
DØTUPD.FTN      /TR:BLØCKS/WR
0001             INTEGER FUNCTIØN CØMPAR(A,I,N,B,J)
0002            IMPLICIT INTEGER (A-Z)
0003            LØGICAL*1 A(1),B(1)
                C
                C CØMPARE CHARACTER STRINGS
                C
                C IF A .LT. B  CØMPAR=-1
                C IF A .EQ. B  CØMPAR=0
                C IF A .GT. B  CØMPAR=+1
                C
0004                IS=1
0005                JS=J
0006                CØMPAR=0
0007                DØ 70 K=1,N
0008                M1=A(IS)
0009                M2=B(JS)
0010                IS=IS+1
0011                JS=JS+1
0012                IF(M1-M2)50,70,60
0013            50  CØMPAR= -1
0014                RETURN
0015            60  CØMPAR= +1
0016                RETURN
0017            70  CØNTINUE
0018                RETURN
0019                END
```

3.11 FUNCTION COMPAR

A flow chart for this function is not available.

3.12 SUBROUTINE SUBSTR

```
FØRTRAN IV-PLUS V02-04          10:15:30    01-AUG-77          PAGE 28
DØTUPD.FTN      /TR:BLØCKS/WR
0001            SUBRØUTINE SUBSTR(A,I,N,B,J,M)
      C MØVE A TØ B
      C A= INPUT ARRAY
      C I = STARTING BYTE PØSITIØN
      C
      C N = NUMBER ØF BYTES TØ MØVE
      C
      C B= RECEIVING ARRAY
      C
      C J= STARTING BYTE PØSITIØN IN RECEIVING ARRAY
      C
      C M = NUMBER ØF BYTES TØ BE RECEIVED
      C
      C NØTE:  TRUNCTED ØR BLANK FILLED IN RECEIVING ARRAY IF
      C        NUMBER ØF BYTES MØVED IS DIFFERENT THAN NUMBER ØF BYTES RECEIVE
      C
      C NØTE:  BYTE PØSITIØN ARE TØ BE CØUNTED STARTING WITH THE NUMBER  1
      C
      C
0002            IMPLICIT INTEGER (A-Z)
0003            LØGICAL*1 A(1),B(1)
0004            DATA BLANK/2H  /
0005            IS=I
0006            JS=J
0007            L=0
0008            IF(N .EQ. 0) GØ TØ 20
0009            L=N
0010            IF( L .GT. M ) L=M
0011            DØ 10 K=1,L
0012            B(JS)=A(IS)
0013            IS=IS + 1
0014            JS=JS + 1
0015       10   CØNTINUE
0016            IF( N .GE. M) RETURN
0017       20   L=L + 1
0018            DØ 30 K=L,M
0019            B(JS)=BLANK
0020            JS=JS+1
0021       30   CØNTINUE
0022            RETURN
0023            END
```

3.12 SUBROUTINE SUBSTR

A flow chart for this subroutine is not available.

3.13 SUBROUTINE OPMESS

```
FØRTRAN IV-PLUS V02-04          10115132     01-AUG-77          PAGE 30
DØTUPD.FIN      /TR:BLØCKS/NR
0001            SUBRØUTINE ØPMESS(DISKNM)
0002            INTEGER DISKNM
0003            LØGICAL*1 ANS(16),BLANK
0004            DATA BLANK /1H /
                C
                C
                C
0005     20     TYPE 30,DISKNM
0006     30     FØRMAT(1HØ,'THE NEXT SEGMENTS TØ BE PRØCESSED WILL BE FØUND ON ',
                *           'DISK NUMBER ',I4,//,1X,'IF YØU WISH TØ CØNTINUE',
                *           'PLEASE MØUNT DISK NUMBER ',I4,/,
                *           'THEN TYPE THE LETTER (C) AND CARRIAGE RETURN.',
                *         1HØ,'IF YØU WISH TØ ABØRT THIS JØB THEN TYPE IN THE ',
                *           'LETTER (X) AND CARRIAGE RETURN',10X,
                *           'THANK YØU')
                C
                C
0007            ACCEPT 40,ANS
0008     40     FØRMAT(16A1)
0009            DØ 50 J=1,16
0010            IF(ANS(J) .NE. BLANK) GØ TØ 60
0011     50     CØNTINUE
0012            GØ TØ 20
0013     60     IF(ANS(J) .EQ. 'Y') RETURN
0014            TYPE 70
0015     70     FØRMAT(1HØ,'ANALYST ABØRTED RUN!')
0016            STØP 'ABØRT'
0017            END
```

3.13 SUBROUTINE OPMESS

A flow chart for this subroutine is not available.

```
HPFORTRAN IV+PLUS V02-04                14102119    28-JUL-77          PAGE 1
BSTAT.FTN       /TR:BLOCKS/WR
C
C
C       CLUSTER STATISTICS UPDATE FROM CAMS/CAS
C         INTERFACE TAPE - BSTAT.FTN - APR 1977
C
C
0001            IMPLICIT INTEGER(A-Z)
0002            INCLUDE 'SYS[300,3]CAMSCOM.INC'
0003  *         INCLUDE 'SYS[300,3]CAMSPARAM.INC'
0004  *         PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      *        1,MAXV=11,NDOTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *        2NOSPWD=6,NODTWD=10
0005  *         EQUIVALENCE (C1,ACDATE),(C2,ISEG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0006  *         INTEGER C1(469),C2(256),C3(71),C4(348),C5(629)
      *  C.
0007  *         INTEGER ACDATE,SUBCAT,SUBPOP,CATKNT,CATTH
0008  *         BYTE CHNVEC,NOCHAN,NOSUB,DOTCAT,DOTCLU
0009  *         COMMON/COM1/ACDATE(2,MAXACC),CHNVEC(MAXCHN,MAXACC),NOCHAN,NOSUB,
      *        1SUBCAT(MAXSUB),SUBPOP(MAXSUB),CATKNT(MAXCAT),CATTH(MAXCAT),NODU,
      *        2NODU,NOTH,DOTCAT(NDOTS),DOTCLU(NDOTS)
      *  C.
0010  *         INTEGER ADATES,SUNAZ,ANALST,FLDDAY,DOTDAY,PDATE1,TDATE1
0011  *         INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RANDOM,GRID
0012  *         BYTE DELFLG,NOACQ,SOILGR,SUNEL,NSTART,NTYPE1,ALP,ALPO
0013  *         BYTE PCTCT,PCTCTO,VAR,VARO,DLABEL,TYPE
0014  *         COMMON/COM2/ISEG,DELFLG,NOACQ,ADATES(2,MAXACD),SOILGR(MAXACD),
      *        1SUNEL(MAXACD),SUNAZ(MAXACD),IMDATE(2),ANALST(5),FLDDAY(2),
      *        2DOTDAY(2),NSTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      *        3PDATE3(2),TDATE3(2),NOCAT,CATNAM(MAXCAT),ALP(MAXCAT),ALPO,
      *        4          PCTCT(MAXCAT),PCTCTO,VAR(MAXCAT),VARO
      *  C.
0015  *         INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      *        1UFLAG4
0016  *         INTEGER PFLAG,DSKMNT
0017  *         COMMON/COM3/PFLAG,DSKMNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
      *        1,UFLAG2,UFLAG3,UFLAG4,NEWLAB(MAXSUB)
      *  C.
0018  *         INTEGER TX1,TY1,TX2,TY2,ACQISP,G,B,DTWIND,DOTARY,GMIN,GMAX,FUL
0019  *         INTEGER SPWIND,CLAWND,CLUWND
0020  *         COMMON/COM4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACQISP(2),III(4),G(4),
      *        1B(4),DTWIND(5,NODTWD),SPWIND(5,NOSPWD),IMWIND(4),NUMDOT,
      *        2DOTARY(NDOTS),GMIN,GMAX,FUL(2,7),CLAWND(8),CLUWND(8)
0021  *         COMMON/COM5/DISKID,RANDOM(NDOTS),GRID(NDOTS),DLABEL(NDOTS),
      *        1TYPE(NDOTS),RECLOC
0022              BYTE IBUF(800)
0023               BYTE V(4)
0024              EQUIVALENCE(X,IGR)
0025              DIMENSION IGR(2)
0026              REAL X
0027              REAL STAT(2160)
0028              BYTE CLASS(6)
0029              DIMENSION T(60),TP(60),S(60)
0030              BYTE BLK(2)
0031              BYTE ALPHA(36)
```

```
0032            BYTE ZER(1)
0033             BYTE LITUIC(13)
0034            BYTE STATN(11)
0035            BYTE DMYR(9)
0036            BYTE HMSEC(9)
0037            EQUIVALENCE(BLCAT,BLK)
0038            BYTE DIREC(24)
0039            BYTE BETA(30)
0040             REAL T1,DELTA
0041            DATA DIREC/'S','Y','0','1','1','3','0','3',',','3','0','1',':','1',
               *'D','I','R','F','I','L','E',',','D','A','T'/
0042            DATA BLK/' ',' '/
0043             DATA STATN/'T','S','T','R','T',',',':','0','/',':','T',':',':','1'/
0044             DATA LITUIC/'D','B','2','1','L','3','0','3',',',':','3','0',':',':'/
0045            DATA ZER0/'0'/
0046            T1=SECNDS(0.0)
        D        IF(1.EQ.1) NMCAT=0
        C
        C READ AND PRØCESS RECOGNITION SEGME T
        C RECØRDS
        C
0047            FILE=0
0048             OPN3=0
0049            R=0
0050            MDFLG=0
0051             IF(1.EQ.1) GØ TØ 79
0052            GØ TØ 80
0053       47   CALL CDREC(IBUF,R,FILE)
0054            IF(R.EQ.0)  CLØSE(UNIT=1)
0055            IF(FILE.GE.2)  GØ TØ 50
0056            R=R+1
0057            IF(IBUF(1).NE.'R')  GØ TØ 47
0058            DECODE(4,104,IBUF(146))  SEGN
0059            NUACQ=0
0060            DID=0
        C
        C SET UP DATA BASE DIRECTORY FILE NAME
        C
0061            N=14
0062            L=1
        D        IF(L.EQ.1)  GØ TØ 41
0063            N=1
0064            DØ 42 I=L,L+12
0065            BETA(I)=DIREC(N)
0066            N=N+1
0067       42   CØNTINUE
0068            L=L+13
0069       41   CØNTINUE
0070            DØ 40 I=L,L+10
0071            BETA(I)=DIREC(N)
0072            N=N+1
0073       40   CØNTINUE
0074            L=L+11
0075            BETA(L)=0
        D        IF(1.EQ.1)  GØ TØ 60
0076            CALL DSKCHK(SEGNØ,PTR,DSKID,FLAG)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
BSTAT,FTN       /TR:BLØCKS/WR
0077            IF(FLAG.EQ.1)  GØ TØ 62
0078            IF(FLAG.EQ.2)  GØ TØ 64
        D       IF(1.EQ.1)  GØ TØ 60
0079            IF(ØPN3.EQ.1) GØ TØ 70
0080            ØPEN(UNIT=3,TYPE='ØLD',NAME=BETA,ACCESS='DIRECT',
               +MAXREC=200,RECØRDSIZE=128)
0081            ØPN3=1
0082    70      CØNTINUE
0083            FIND(3'PTR)
0084            READ(3'PTR)  C2
0085    60      CØNTINUE
        C
        C SET UP STAT FILE NAME:  ( ? )XXXXTSTAT.DAT
        C
0086            L=1
0087            N=1
        D       IF(N.EQ.1)  GØ TØ 14
0088             DØ 12 I=L,L+12
0089             ALPHA(I)=LITUIC(N)
0090             N=N+1
0091    12       CØNTINUE
0092             L=L+13
0093             N=1
0094    14      CØNTINUE
0095            DØ 9 I=L,L+3
0096            ALPHA(I)=IBUF(147+N)
0097             IF(ALPHA(I).EQ.' ')  ALPHA(I)='0'
0098            N=N+1
0099      9     CØNTINUE
0100            L=L+4
0101            N=1
0102            DØ 11 I=L,L+10
0103            ALPHA(I)=STATN(N)
0104            N=N+1
0105      11    CØNTINUE
0106            L=L+11
0107            ALPHA(L)=0
        C
        C CØNVERT AND STØRE ACQUITIØN DATES
        C
0108            RI=74
0109            DECØDE(2,102,IBUF(RI))  CYR
0110            DECØDE(3,103,IBUF(RI+2))  CDY
0111            TDATE3(1)=CYR
0112            TDATE3(2)=CDY
0113            RI=154
0114            DØ 1 I=1,4
0115            IF(IBUF(RI).EQ.'0')  GØ TØ 10
0116            DECØDE(2,102,IBUF(RI)) ACDATE(1,I)
0117      102   FØRMAT(I2)
0118            DECØDE(3,103,IBUF(RI+2))  ACDATE(2,I)
0119      103   FØRMAT(I3)
0120            RI=RI+5
0121            GØ TØ 1
0122      10    ACDATE(1,I)=0
0123            ACDATE(2,I)=0
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
0124            IF(NUACQ.EQ.0)  NUACQ=1-1
0125         1  CONTINUE
                C
                C CONVERT AND STORE CHANNEL VECTOR
0126            NØCHAN= 0
0127            JJ=0
0128            DØ 97 IA=1,4
0129            DØ 98 IC=1,4
0130            JJ=JJ+1
0131             CHNVEC(IC,IA)=0
0132            IF(IBUF(83+JJ).EQ.'1')  GØ TØ 96
0133            GØ TØ 98
0134         96 CHNVEC(IC,IA)=1
0135            NØCHAN=NØCHAN+1
0136         98 CONTINUE
0137         97 CONTINUE
                C
                C
0138            DECØDE(4,104,IBUF(56)) ALSETS
0139        104 FØRMAT(I4)
0140            DECØDE(5,105,IBUF(64)) NØ?Ø
0141        105 FØRMAT(I5)
0142            DECØDE(5,105,IBUF(69)) NØØL
                C
                C
0143             J=1
0144            I=1
0145            SETSR=14
0146            RJ=249
0147            LFLD=32
0148             DØ 8 KK=1,MAXCAT
0149             CATKNT(KK)=0
0150             CATTH(KK)=0
0151         8   CONTINUE
0152             NØTH=0
0153             KK=0
                C
                C PROCESS SUBCLASS RELATED CLASSIFICATION
                C RESULTS
                C
0154         20 K=1
0155         22 CONTINUE
0156             DØ 2 N=1,6
0157         2  CLASS(N)= IBUF(RJ-1+N)
                C
                C SKIP SUBCLASS IF LETTERS 3,4 OF
                C SUBCLASS NAME ARE NØT 'CL'
                C
0158            IF(CLASS(3).NE.'C')  GØ TØ 3
0159            IF(CLASS(4).NE.'L')  GØ TØ 3
                C
                C CØNVERT LAST TWØ CHARACTERS ØF SUBCLASS
                C NAME = STØRE AS SURCLASS NØ.
                C
0160             DECØDE(2,102,CLASS(5))  ISUBCL
0161            BLK(1)=CLASS(1)
```

ORIGINAL PAGE IS
OF POOR QUALITY


```
BSTAT,FTN       /TR:BLØCKS/WR
0162            IF(NØCAT.EQ.0)  GØ TØ 6
0163            DØ  5 KK=1,NØCAT
0164            IF(CATNAM(KK).EQ.BLCAT)  GØ TØ 7
0165       5    CØNTINUE
0166       6    NØCAT=NØCAT+1
0167             KK=NØCAT
0168            CATNAM(NØCAT)=BLCAT
0169       7    CØNTINUE
0170            DECØDE(5,105,IBUF(RJ+22)) SUBCT
0171            DECØDE(5,105,IBUF(RJ+27)) THRES
0172             CATKNT(KK)=CATKNT(KK)+SUBCT
0173             CATTH(KK)=CATTH(KK)+THRES
0174             NØTH=NØTH+THRES
0175             SUBCAT(ISUBCL)=KK
0176            I=I+1
0177       3    IF(J.EQ.ALSETS)  GØ TØ 45
0178            J=J+1
0179            IF(K.EQ.SETSR)  GØ TØ 13
0180            K=K+1
0181            RJ=RJ+LFLD
0182            GØ TØ 22
              C
              C READ NEXT RECØGNITIØN SEGMENT RECØRD
              C
0183         13 CALL CDRED(IBUF,R,FILE)
0184            RJ=3
0185            SETSR=22
0186            GØ TØ 20
              C
              C
0187         45 CØNTINUE
0188             I=I-1
0189            NØCLUS=I
0190             IF(NØCLUS.EQ.0)  GØ TØ  66
0191            NØSUB=NØCLUS
              C (SET NØSUB)
              C
              C READ AND PRØCESS SUBCLASS STATISTICS
              C RECØRD
0192         44  CØNTINUE
              C
0193            CALL CDRED(IBUF,R,FILE)
0194             IF(IBUF(1).NE.'S')  GØ TØ 44
0195            DECØDE(2,102,IBUF(10)) NCH
0196            DECØDE(3,103,IBUF(5))  STSETS
              C (INSERT TEST!  IS ALSETS = STSETS
0197            NØT=NCH/4
0198            J=1
0199            I=1
0200            LFLD=11+9*NCH
0201         37 RJ=12
0202            DECØDE(2,102,IBUF(8))  SETSR
0203            K=1
              C
              C ØBTAIN STATISTICS FØR THE J TH SUBCLASS
              C AS INPUT FRØM TAPE
```

```
BSTAT,FTN       /TR:BLØCKS/WR
      C
0204    36    CØNTINUE
0205          IF(IBUF(RJ+2),NE,'C')  GØ TØ 4
0206          IF(IBUF(RJ+3),NE,'L')  GØ TØ 4
0207          DECØDE(2,102,IBUF(RJ+4)) INTGR
      C
0208          DECØDE(5,105,IBUF(RJ+6)) SURFACE(I)
0209          L=36*(I-1)+1
      C
      C GET AND CØNVERT STATISTICS IN GRØUPS
      C ØF 4 - A TETRAD ØF CHANNELS
0210          IT=1
0211          ØFF=11
0212    29    CØNTINUE
0213          DØ 30 JJ=1,4
0214          DECØDE(5,105,IBUF(RJ+ØFF))  MEAN
0215          STAT(L)=(FLØAT(MEAN))/100.
0216           V(JJ)=STAT(L)
0217          DECØDE(4,104,IBUF(RJ+ØFF+5))  ISD
0218          STAT(L+1)=(FLØAT(ISD))/10.
0219          ØFF=ØFF+9
0220          L=L+2
0221    30    CØNTINUE
      C
      C INSERT GREENESS CALCULATIØNS
      C
0222          IGR(1)=1
0223          IGR(2)=2
0224           CALL KAUTH(V,IGR(1),IGR(2))
0225          STAT(L)=X
0226          L=L+1
0227          IF(IT,EQ,NØT)  GØ TØ 33
0228          IT=IT+1
0229          GØ TØ 29
0230    33    I=I+1
0231    4     IF(J,EQ,STSETS)  GØ TØ 34
0232          J=J+1
0233          IF(K,EQ,SETSR)  GØ TØ 35
0234          K=K+1
0235          RJ=RJ+LFLD
0236          GØ TØ 36
0237    35    CALL CDRED(IBUF,R,FILE)
0238          GØ TØ 37
0239    34    CØNTINUE
      C
      C ØUTPUT XXXXTSTAT FILE TØ DISK
      C
0240           ØPEN(UNIT=4,TYPE='NEW',NAME=ALPHA,ACCESS='SEQUENTIAL',
              FORM='UNFØRMATTED')
0241          WRITE(4)  (C1(I),I=1,469)
0242          K=1
0243    48     CØNTINUE
0244          L=36*(K-1)+1
0245          L2=L+35
0246          J=K+1
0247          WRITE(4)  (STAT(II),II=L,L2)
```

```
0STAT.FTN       /TR:BL@CKS/WR
0248            IF(K.EQ.N@CLUS)  G@ T@ 49
0249            K=K+1
0250            G@ T@ 48
0251      49    C@NTINUE
0252            CL@SE(UNIT=4)
0253            IF(HDFLG.EQ.1)  G@ T@ 56
0254      79     C@NTINUE
0255      80    C@NTINUE
0256            WRITE(6,50)
0257      50    F@RMAT('1',10X,'CAMS I=100 DATA BASE TRANSACTI@N REP@RT'//)
0258            CALL DATE(DMYR)
0259            WRITE(6,51)  DMYR
0260      51    F@RMAT(50X,'DATE  ',9A1)
0261            CALL TIME(HMSEC)
0262            WRITE(6,52)  HMSEC
0263      52    F@RMAT(50X,'TIME  ',8A1)
0264             WRITE(6,77)
0265      77     F@RMAT('  ')
0266            WRITE(6,53)
0267      53    F@RMAT(15X,'ERIPS CAMS/CAS INTERFACE UPDATE')
0268             WRITE(6,77)
0269            WRITE(6,54)
0270      54    F@RMAT(17X,'DISK    DATE @F       ACQUISITI@NS USED')
0271            WRITE(6,55)
0272      55    F@RMAT(8X,'SEGMENT   ID  CLASSIFICATI@N DATE 1 DATE 2 DATE 3',
     *' DATE 4')
0273             WRITE(6,77)
0274            HDFLG=1
0275             G@ T@ 47
0276      56    C@NTINUE
0277             WRITE(6,57)  SEGN@,DSKID,CYR,CDY,
     *(ACDATE(1,I),ACDATE(2,I),I=1,NUACQ)
0278      57     F@RMAT(10X,I4,3X,I2,6X,I2,1X,I3,6X,4(I2,1X,I3,2X))
0279             WRITE(6,77)
          D     IF(1.EQ.1)  G@ T@ 47
0280      75    C@NTINUE
0281            WRITE(3'PTR)  C2
0282            G@ T@ 47
0283      58    C@NTINUE
0284            DELTA=SECNDS(T1)
0285      59    F@RMAT('0',10X,'ELAPSED TIME= ',F5.1,2X,'SEC')
0286             R=-1
0287             CALL CDRED(IBUF,R,FILE)
          D     IF(1.EQ.1)  ST@P
0288             CL@SE(UNIT=3)
0289            CL@SE(UNIT=6)
0290             ST@P
0291      61    C@NTINUE
0292            FLAG=0
0293            CALL CDRED(IBUF,R,FILE)
0294            IF(FILE.GE.2)  G@ T@ 58
0295            IF(IBUF(1).EQ.'R')  G@ T@ 61
0296            G@ T@ 47
0297      62    C@NTINUE
0298            WRITE(6,63)  SEGN@,DSKID
0299      63    F@RMAT(10X,I4,3X,I2,4X,'THIS SEGMENT N@T @N CURRENT DISK')
```

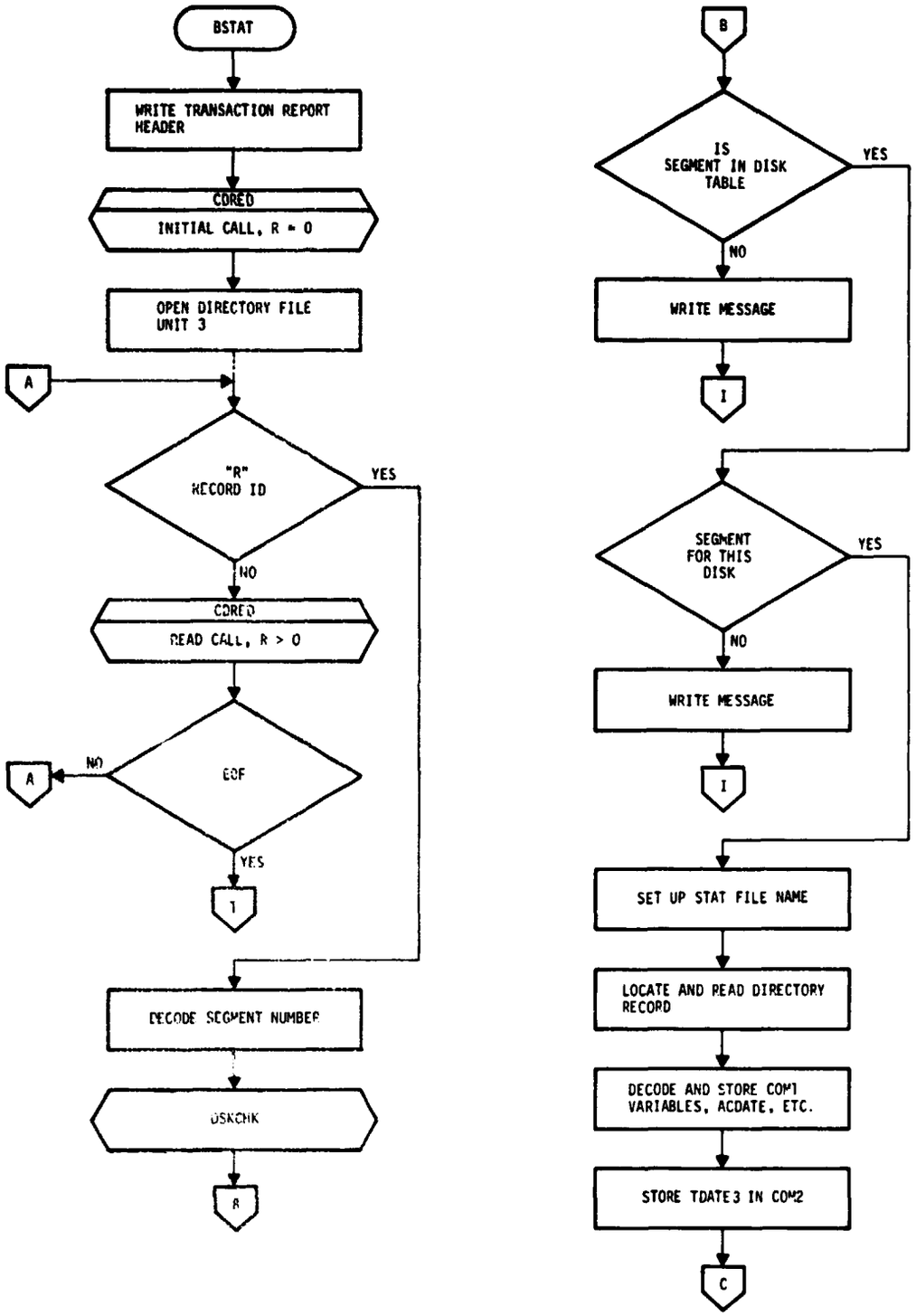
```
FØRTRAN IV-PLUS V02-04          14:02:19    25-JUL-77          PAGE 8
BSTAT.FTN       /TR:BLØCKS/WR
0300            GØ TØ 61
0301       64   CØNTINUE
0302            WRITE(6,65)  SEGNØ
0303       65   FØRMAT(10X,I4,10X,'THIS SEGMENT NOT IN DISK TABLE')
0304             GØ TØ 61
0305       66    CØNTINUE
0306             WRITE(6,67)  SEGNØ,DSKID
0307       67    FØRMAT(10X,I4,3X,I2,4X,'STATS NOT LØADED- NØ CLUSTERS THIS',
                *' SEGMENT')
0308            GØ TØ 61
0309            END
```

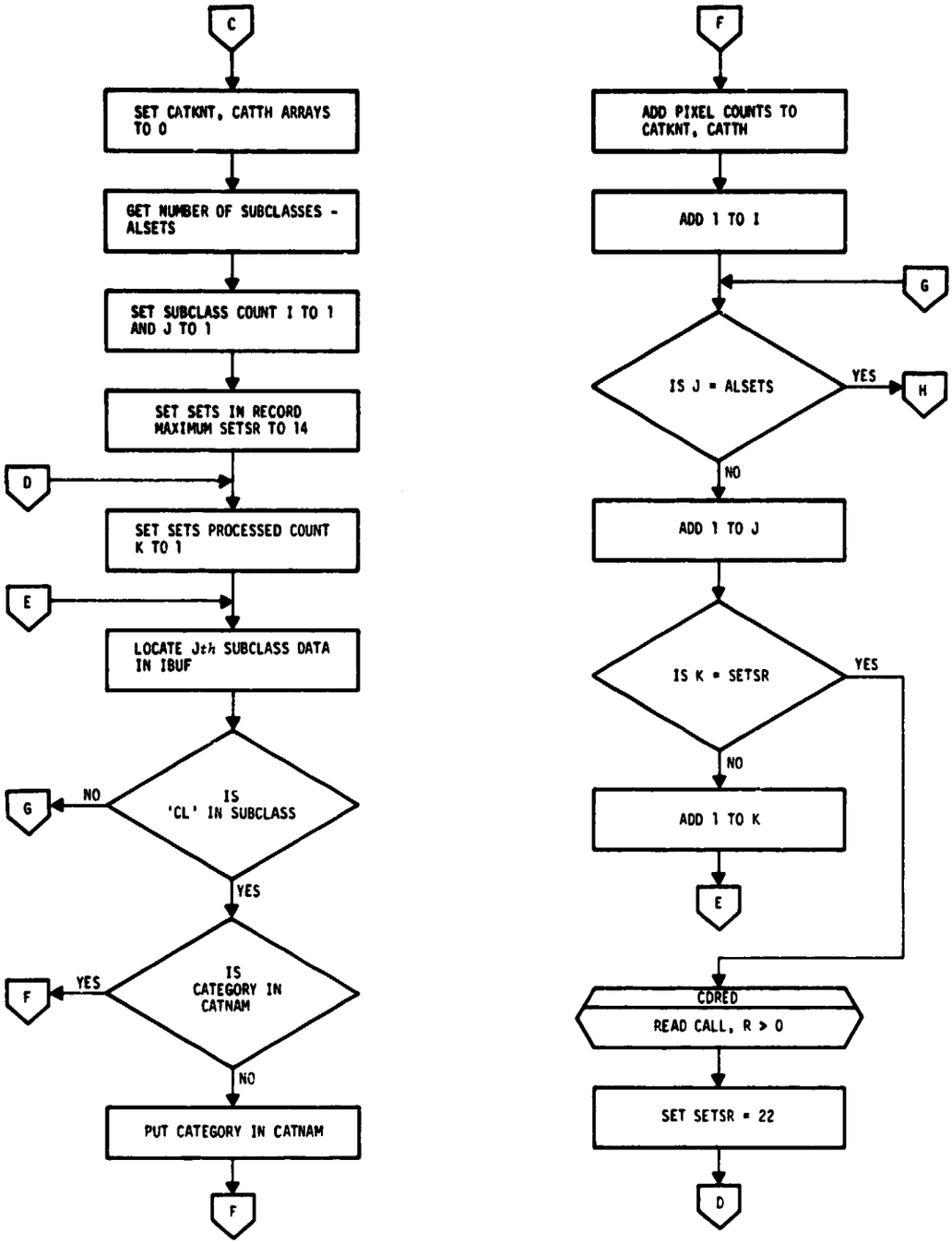
ORIGINAL PAGE IS
OF POOR QUALITY

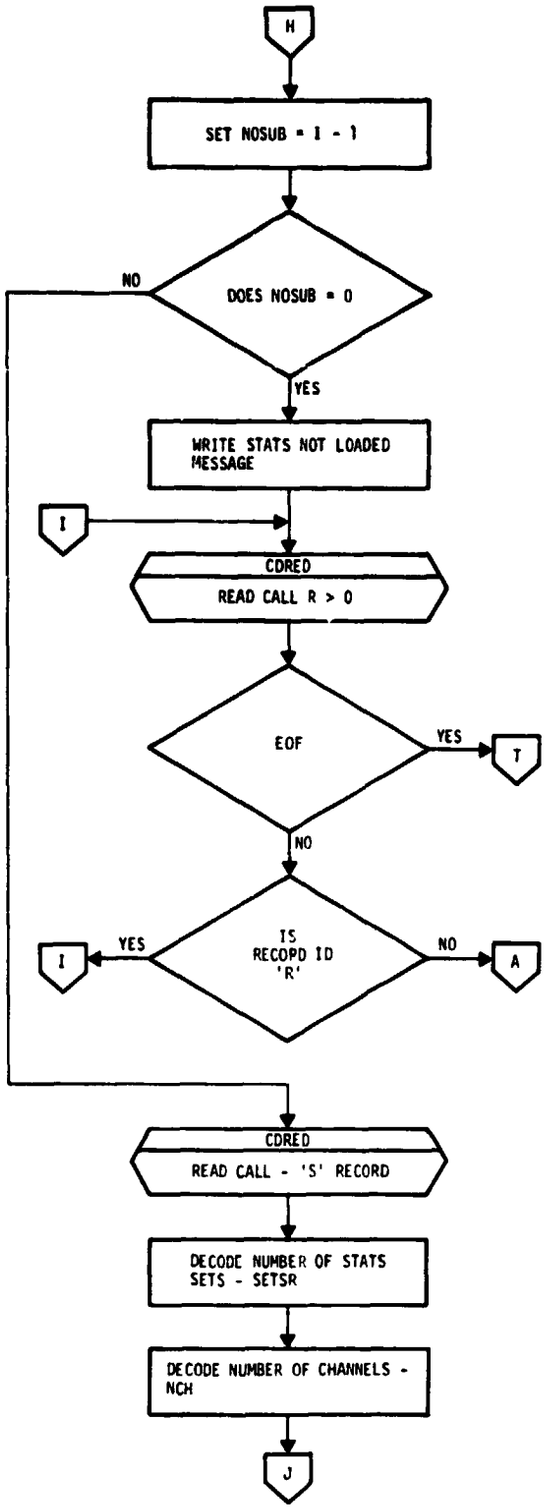
4.1 SUBROUTINE CDRED

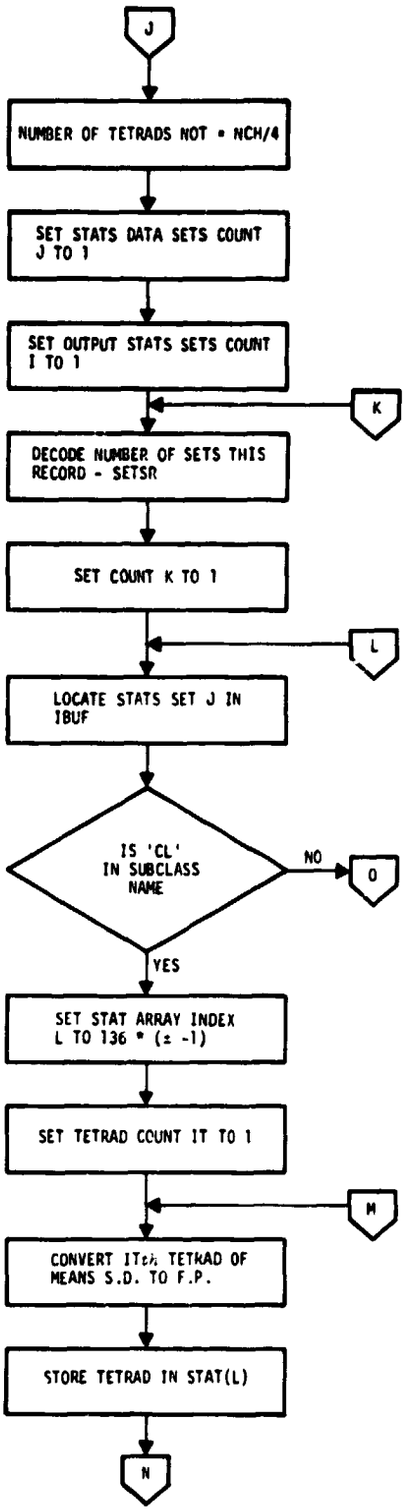
```
HF0RTRAN IV-PLUS V02-04          14:03:14   28-JUL-77          PAGE 1
CDRED.FTN       /TRIBL0CKS/WR
0001            SUBR0UTINE CDRED(IBUF,R,FILE)
0002            IMPLICIT INTEGER(A-Z)
0003            INTEGER*2 ISTAT(2),IPRM(6)
0004             EQUIVALENCE(ISTAT(1),IB)
0005             BYTE IB(2)
0006            L0GICAL*1 IBUF(1)
0007            DIMENSI0N IWDAT(1)
0008            DIMENSI0N IA(37)
0009            DATA XDEV/2HXT/
0010            DATA MDEV/2HMT/
0011            REC0RD=R
0012            IF(REC0RD.GT.0)  G0 T0 10
0013             ERCNT=0
0014             IF(REC0RD.EQ.-1)  G0 T0 6
0015     12      C0NTINUE
0016            FILE=0
         D      WRITE(5,49)
0017     49     F0RMAT(10X,' TYPE M 0R X F0R TAPE DEVICE C0DE'/)
0018            READ(1,51,END=99)  IA
0019            IF(IA(1).EQ.'M')  G0 T0 13
0020            IF(IA(1).EQ.'X')  G0 T0 14
0021     99     C0NTINUE
0022            WRITE(6,97)
0023     97     F0RMAT(10X,' CARD ERR0R  '//)
0024            WRITE(6,97)
0025            STOP
0026     13     IDEV=MDEV
0027            G0 T0 15
0028     14     IDEV=XDEV
0029     15     C0NTINUE
         D      WRITE(5,50)
0030     50     F0RMAT(10X,' TYPE TAPE UNIT NUMBER - 0 0R 1')
0031            READ(1,51,END=99)  IA
0032     51     F0RMAT(37A2)
0033             IF(IA(1).EQ.'0')  G0 T0 18
0034             IF(IA(1).EQ.'1')  G0 T0 19
0035             G0 T0 99
0036     18      IUNT=0
0037             G0 T0 20
0038     19      IUNT=1
0039     20      C0NTINUE
0040            ILUN=9
0041            IDS=0
0042            ISR=0
0043            CALL ASNLUN(ILUN,IDEV,IUNT,IDS)
0044            IF(IDS.LT.0)  G0 T0 1
0045            CALL GETADR(IPRM,IBUF)
0046     6       C0NTINUE
0047            CALL Q10(%2400,ILUN,1,,ISTAT,IPRM,ISR)
0048            IF(ISR.LT.0)  G0 T0 2
0049             IF(REC0RD.EQ.-1)  G0 T0 11
0050            IPRM(2)=3600
0051     10     C0NTINUE
0052            ISW=0
0053            REC0RD=REC0RD+1
0054            CALL Q10(%1000,ILUN,1,,ISTAT,IPRM,ISW)
```

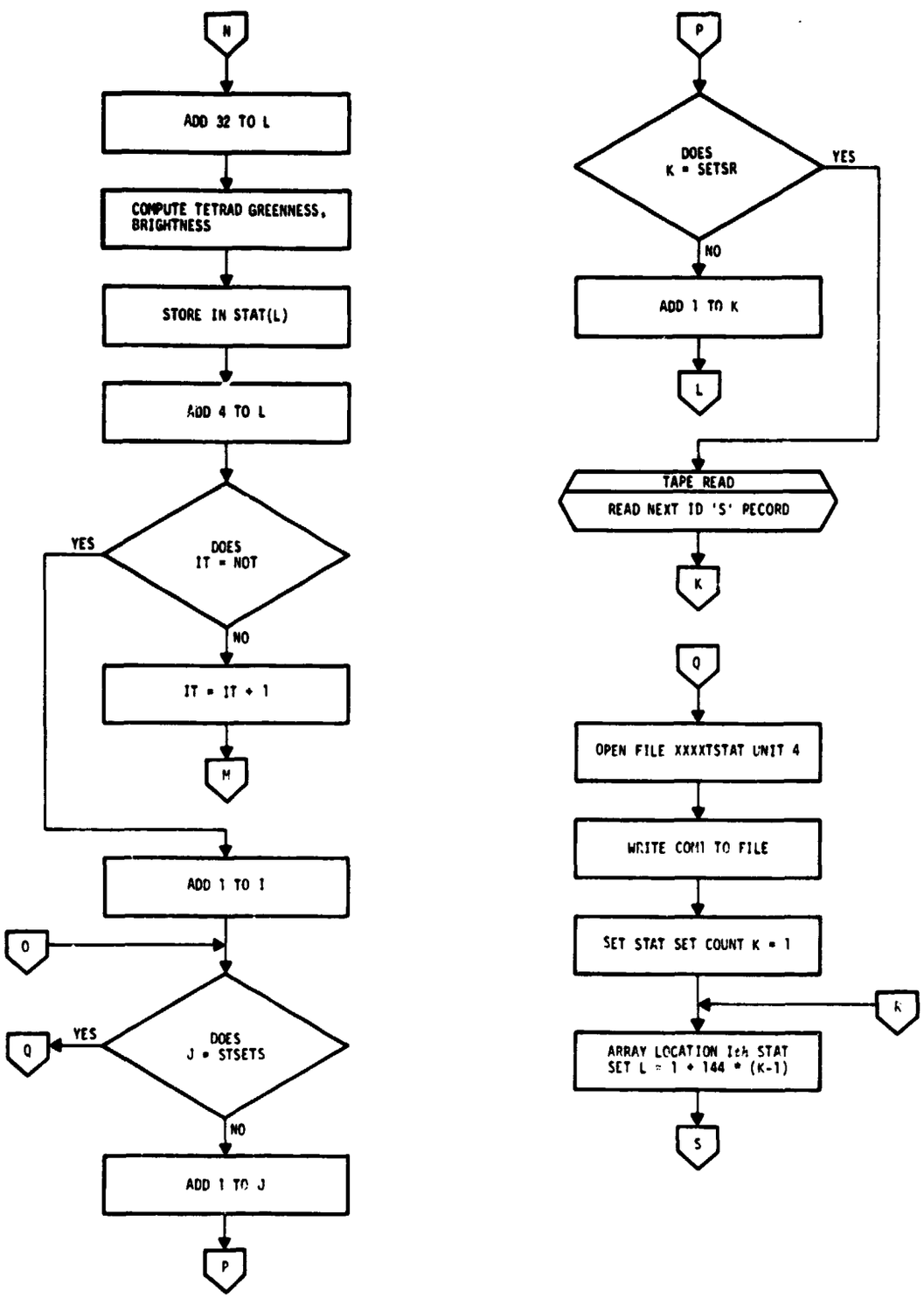
```
CDRED,FTN       /TR:BLØCKS/WR
0055            IF(ISW.LT.0)  GØ TØ 3
0056            CALL WAITFR(1,IDS)
0057            IF(IDS.LT.0)  GØ TØ 4
        C        WRITE(6,101) (IB(I),I=1,2),ISTAT(2)
0058    101      FØRMAT(1HØ,10X,2I4,3X,I5,2X,'BYTES TRANSFERED')
0059             ERCØDE=IB(1)
0060            IF(ERCØDE.LT.0)  GØ TØ 5
0061            RETURN
0062    1       CØNTINUE
0063            WRITE(6,100)  IDS
0064    100     FØRMAT(1H ,'  ASLUN CALL DSW=  ',I6)
0065            STØP
0066    2       CØNTINUE
0067            WRITE(6,200) ISR
0068    200     FØRMAT(1H ,'REWIND DSW = ',I6)
0069            STØP
0070    3       CØNTINUE
0071            WRITE(6,300)  ISW
0072    300     FØRMAT(1H ,'  READ QIØ DSW = ',I6)
0073            STØP
0074    4       CØNTINUE
0075            WRITE(6,400)  IDS
0076    400     FØRMAT(1H ,'  WAIT DSW = ',I6)
0077            STØP
0078    5       CØNTINUE
0079            IF(ERCØDE.EQ.-10)  FILE=FILE+1
0080            IF(ERCØDE.EQ.-10)  RETURN
0081             IF(ERCØDE.NE.-4)  GØ TØ 7
0082             ERCNT=ERCNT+1
0083             RETURN
0084    11       CØNTINUE
0085             WRITE(6,700)
0086    700      FØRMAT('  ')
0087             IF(ERCNT.EQ.0)  RETURN
0088             WRITE(6,600)  ERCNT
0089    600      FØRMAT('1',10X,' TAPE ERRØRS ENCØUNTERED = ',I5)
0090             RETURN
0091    7        CØNTINUE
0092            WRITE(6,500)  ERCØDE
0093    500     FØRMAT(1HØ,'  I/Ø STATUS BLØCK ERRØR CØDE= ',I6)
0094            STØP
0095            END
```

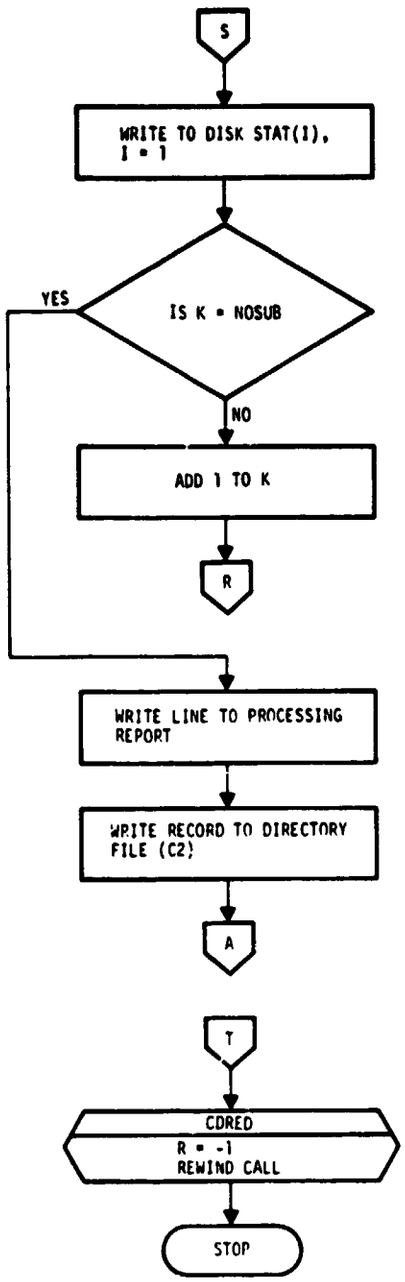

BSTAT
WRITE TRANSACTION REPORT HEADER
CORED
INITIAL CALL, R = 0
OPEN DIRECTORY FILE UNIT 3
A
"R" RECORD ID
YES
NO
CORED
READ CALL, R > 0
EOF
NO
A
YES
T
DECODE SEGMENT NUMBER
DSKCHK
B
IS SEGMENT IN DISK TABLE
YES
NO
WRITE MESSAGE
I
SEGMENT FOR THIS DISK
YES
NO
WRITE MESSAGE
I
SET UP STAT FILE NAME
LOCATE AND READ DIRECTORY RECORD
DECODE AND STORE COM1 VARIABLES, ACDATE, ETC.
STORE TDATE3 IN COM2
C


C
SET CATKNT, CATTH ARRAYS TO 0
GET NUMBER OF SUBCLASSES - ALSETS
SET SUBCLASS COUNT I TO 1 AND J TO 1
SET SETS IN RECORD MAXIMUM SETSR TO 14
D
SET SETS PROCESSED COUNT K TO 1
E
LOCATE Jth SUBCLASS DATA IN IBUF
IS 'CL' IN SUBCLASS
NO
G
YES
IS CATEGORY IN CATNAM
YES
F
NO
PUT CATEGORY IN CATNAM
F
F
ADD PIXEL COUNTS TO CATKNT, CATTH
ADD 1 TO I
G
IS J = ALSETS
YES
H
NO
ADD 1 TO J
IS K = SETSR
YES
NO
ADD 1 TO K
E
CDRED
READ CALL, R > 0
SET SETSR = 22
D


H
SET NOSUB = I - 1
DOES NOSUB = 0
NO
YES
WRITE STATS NOT LOADED MESSAGE
I
CDRED
READ CALL R > 0
EOF
YES
T
NO
IS RECOPD ID 'R'
YES
I
NO
A
CDRED
READ CALL - 'S' RECORD
DECODE NUMBER OF STATS SETS - SETSR
DECODE NUMBER OF CHANNELS - NCH
J


J
NUMBER OF TETRADS NOT = NCH/4
SET STATS DATA SETS COUNT J TO 1
SET OUTPUT STATS SETS COUNT I TO 1
K
DECODE NUMBER OF SETS THIS RECORD - SETSR
SET COUNT K TO 1
L
LOCATE STATS SET J IN IBUF
IS 'CL' IN SUBCLASS NAME
NO
O
YES
SET STAT ARRAY INDEX L TO 136 * (I -1)
SET TETRAD COUNT IT TO 1
M
CONVERT ITth TETRAD OF MEANS S.D. TO F.P.
STORE TETRAD IN STAT(L)
N


N
ADD 32 TO L
COMPUTE TETRAD GREENNESS, BRIGHTNESS
STORE IN STAT(L)
ADD 4 TO L
DOES IT = NOT
YES
NO
IT = IT + 1
M
ADD 1 TO I
O
DOES J = STSETS
Q
YES
NO
ADD 1 TO J
P
P
DOES K = SETSR
YES
NO
ADD 1 TO K
L
TAPE READ
READ NEXT ID 'S' RECORD
K
Q
OPEN FILE XXXXTSTAT UNIT 4
WRITE COM1 TO FILE
SET STAT SET COUNT K = 1
R
ARRAY LOCATION Ith STAT
SET L = 1 + 144 * (K-1)
S


S
WRITE TO DISK STAT(I),
I = 1
YES
IS K = NOSUB
NO
ADD 1 TO K
R
WRITE LINE TO PROCESSING
REPORT
WRITE RECORD TO DIRECTORY
FILE (C2)
A
T
CDRED
R = -1
REWIND CALL
STOP

4.2 SUBROUTINE IOPRNT

```
MFØRTRAN IV-PLUS V02-04          12:49:51   31-AUG-77          PAGE 1
IØPRNT.FTN       /TRACKS/WR
0001             SUBROUTINE IØPRNT(IWØ,Iw1,ERCØDE)
0002             IMPLICIT INTEGER(A-Z)
0003             LØGICAL*1 IWØ(1)
                 WRITE(9,100) (IwØ(I),I=1,2),Iw1
0004     100     FØRMAT(1HØ,2I5,Ø1Ø)
0005             ERCØDE=IWØ(1)
0006             RETURN
0007             END
```

4.2 SUBROUTINE IOPRINT

A flow chart for this subroutine is not available.

ORIGINAL PAGE IS
OF POOR QUALITY

4.3 SUBROUTINE HEXD

```
HEXD.FTN        /TR:BLOCKS/WR
0001            SUBROUTINE HEXD(BUF,IB)
0002            IMPLICIT INTEGER(A-Z)
0003            LOGICAL*1  BUF(1)
0004            DIMENSION ISYM(16)
0005            LOGICAL*1 IC(2)
0006            DIMENSION IPRINT(108)
0007             EQUIVALENCE(IU,IC(1))
0008            DATA ISYM/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,1HA,1HB,
              * 1HC,1HD,1HE,1HF/
0009            DATA IBLANK/1H /
0010            K=1
0011            IX=MOD(IB,36)
0012            NL=IB/36
0013            IF(NL.EQ.0)  GO TO 10
0014            DO  2  IP=1,108
0015     2      IPRINT(IP)=IBLANK
0016            NC=108
0017     8      CONTINUE
0018            DO  4  IL=1,NL
0019            DO  1  IP=1,NC
0020            IF(MOD(IP,3).EQ.0)  GO TO 6
0021            IF(MOD(IP,3).EQ.2)  GO TO 3
0022            IC(1)=BUF(K)
0023            L=IU/16
0024             IF(L.LT.0) L=-L
0025            GO TO 5
0026     3      L=MOD(IU,16)
0027     5      L=L+1
0028             IF(L.LT.0) L=-L
0029            IPRINT(IP)=ISYM(L)
0030            GO TO 1
0031     6      K=K+1
0032     1      CONTINUE
0033            WRITE(6,7)  (IPRINT(J),J=1,108)
0034     4      CONTINUE
0035     7       FORMAT(1H0,108A1)
0036            IF(IX.EQ.0)  GO TO 15
0037     10     CONTINUE
0038            IF(IX.EQ.0)   GO TO 15
0039            NL=1
0040            NC=3*IX
0041            IX=0
0042            DO  11  IP=NC,108
0043     11     IPRINT(IP)=IBLANK
0044            GO TO 8
0045     15     CONTINUE
0046            RETURN
0047            END
```

ORIGINAL PAGE IS
OF POOR QUALITY

4.3 SUBROUTINE HEXD

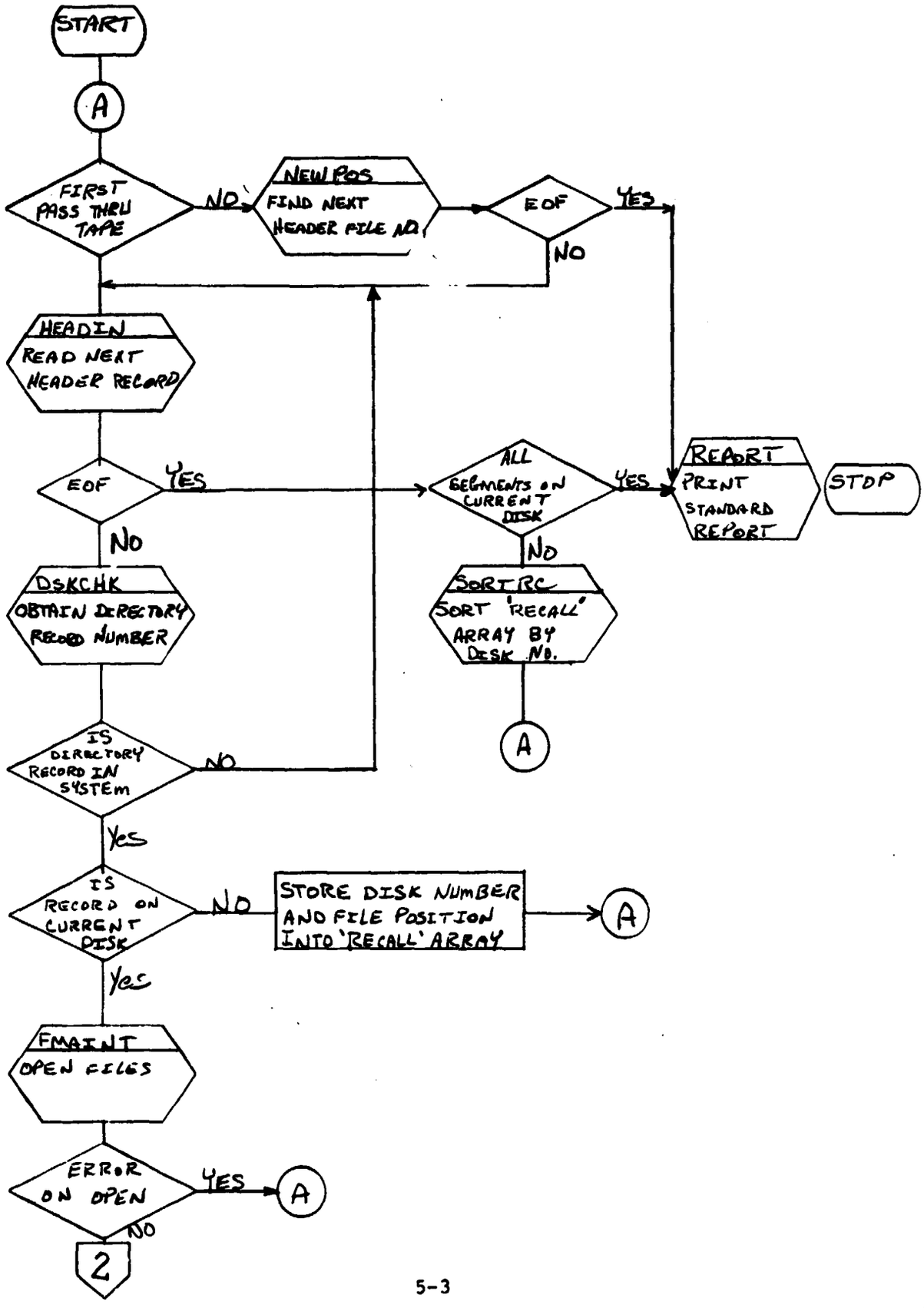
A flow chart for this subroutine is not available.

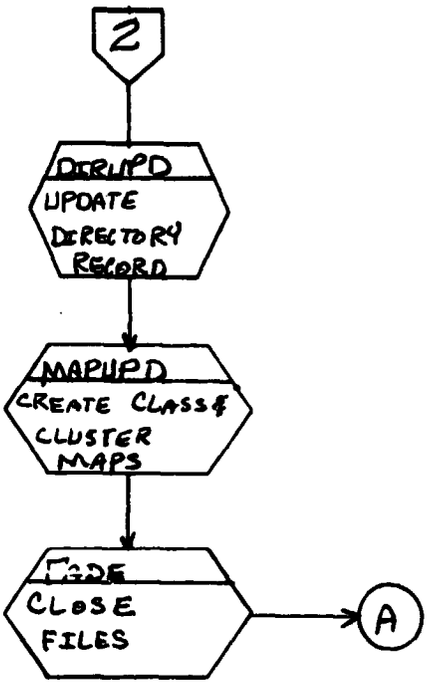
5. CLASSIFICATION AND CLUSTER MAP UPDATE PROGRAM DTERM

```
HFØRTRAN IV-PLUS V02-04                    10:19:33    01-AUG-77          PAGE 1
DTERM,FTN        /TR:BLØCKS/WR
0001               PRØGRAM DTERM
0002               IMPLICIT INTEGER(A-Z)
0003               INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0004  *          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      *         1,MAXV=11,NDØTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *         2NØSPWD=6,NØDTWD=10
0005               DIMENSIØN RECALL(2,30)
0006               DIMENSIØN AID(30),ASEG(30),ADSK(30),AYR(30),AJUL(30)
0007               REAL SEC
0008               DATA LUN/10/
0009               SEC=SECNDS(0.0)
0010               CALL ERRSET(29,,TRUE.,,FALSE.,,TRUE.,,FALSE.,,60)
0011               CNT=0
0012               FILNUM=0
0013               FILPØS=0
           C
           C
           C
0014               NPIX4=NPIX/4
0015       25      CALL HEADIN(ID,SEGNUM,JULDAT,YR,EØF)
0016               IF(EØF .EQ. 0) GØ TØ 300
0017               FILPØS=FILPØS+1
0018               CALL DSKCHK(SEGNUM,PTR,DSKID,FLAG)
0019               IF(FLAG .NE. 0) GØ TØ(210,25),FLAG
           C
0020               CALL FRAINT(SEGNUM,NPIX4,NLIN,ERRFLG)
0021               IF(ERRFLG .EQ. 1) GØ TØ 25
0022               FILNUM=FILNUM+1
           C
           C
0023               AID(FILNUM)=ID
0024               ASEG(FILNUM)=SEGNUM
0025               ADSK(FILNUM)=DSKID
0026               AYR(FILNUM)=YR
0027               AJUL(FILNUM)=JULDAT
           C
           C
           C
0028       80      CALL DIRUPD(YR,JULDAT,ID,PTR)
           C
           C
0029               CALL MAPUPD(ID)
0030               CALL FADE
0031               GØ TØ 25
           C
           C
0032       210     IF(SWTCH .EQ.2) STØP 'ABØRT'
0033               CNT=CNT+1
0034               RECALL(1,CNT)=FILPØS
0035               RECALL(2,CNT)=DSKID
0036               GØ TØ 25
           C
           C
           C
0037       300     IF(CNT .EQ. 0) GØ TØ 400      END ØF JØB --- WITH NØ JØBS AT THIS
0038               CALL SØRTRC(RECALL,CNT)
```

```
FØRTRAN IV-PLUS V02-04          10:19:33    01-AUG-77          PAGE 2
DTERM.FTN       /TR:BLØCKS/WR
0039                TYPE 305
0040     305        FØRMAT(1HØ,'THE FØLLØWING IS A LIST ØF SEGMENTS AND THEIR',/,
                  *  ' DISK NUMBERS THAT WERE NØT PRØCESSED:')
0041                TYPE 310,((RECALL(I,J),I=1,2),J=1,CNT)
0042     310        FØRMAT(1HØ,(1X,2I5))
                    C
                    C
                    C
0043     400        CALL REPØRT(AID,ASEG,ADSK,AYR,AJUL,FILNUM,SEC)
0044                END
```

5. CLASSIFICATION AND CLUSTER MAP UPDATE PROGRAM DTERM




2
DIRUPD
UPDATE
DIRECTORY
RECORD
MAPUPD
CREATE CLASS &
CLUSTER
MAPS
CLOSE
FILES
A

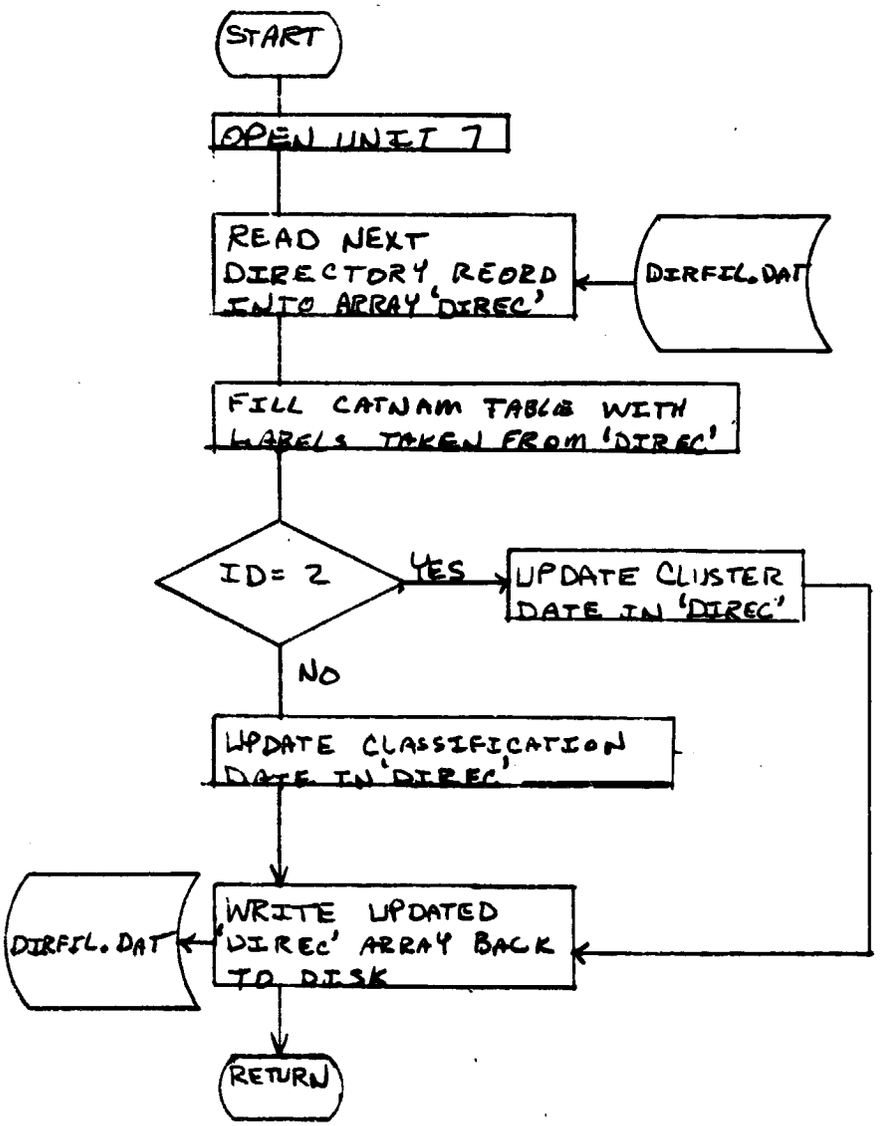
5.1 SUBROUTINE DIRUPD

```
FØRTRAN IV+PLUS V02-04          10:19:43    01-AUG-77          PAGE 4
DTERM.FTN       /YR:6L0CKS/WR
0001            SUBR0UTINE DIRUPD(YR,JULDAT,ID,PTR)
0002            IMPLICIT INTEGER(A-Z)
0003            C0MM0N /CATNAM/ CATNAM(61)
0004            L0GICAL*1 DIRL0G(512)
0005            DIMENSI0N DIREC(256)
0006            EQUIVALENCE(DIREC,DIRL0G)
0007            0PEN(UNIT=7,NAME='[300,300]DIRFILE.DAT;1',TYPE='0LD',
                +  ACCESS='DIRECT',REC0RDSIZE=128,MAXREC=200,ASS0CIATEVARIABLE=I)
                C
0008            FIND(7'PTR)
0009            READ(7'PTR) DIREC
                C
0010            K=0
0011            D0 50 I=52,112
0012            K=K+1
0013      50    CATNAM(K)=DIREC(I)
0014            G0 T0(160,170),ID
0015     160    DIREC(41)=YR
0016            DIREC(42)=JULDAT
0017            G0 T0 180
0018     170    DIREC(45)=YR
0019            DIREC(46)=JULDAT
0020     180    WRITE(7'PTR) DIREC
0021            CL0SE(UNIT=7,DISP0SE='SAVE')
0022            RETURN
0023            END
```

5.1 SUBROUTINE DIRUPD

YR - YEAR
JULDAT - JULIAN DATE
ID - IDENIFICATION OF DATA
1 = CLASSIFICATION
2 = CLUSTER

PTR - POINTER TO NEXT RECORD IN DIRECTORY FILE

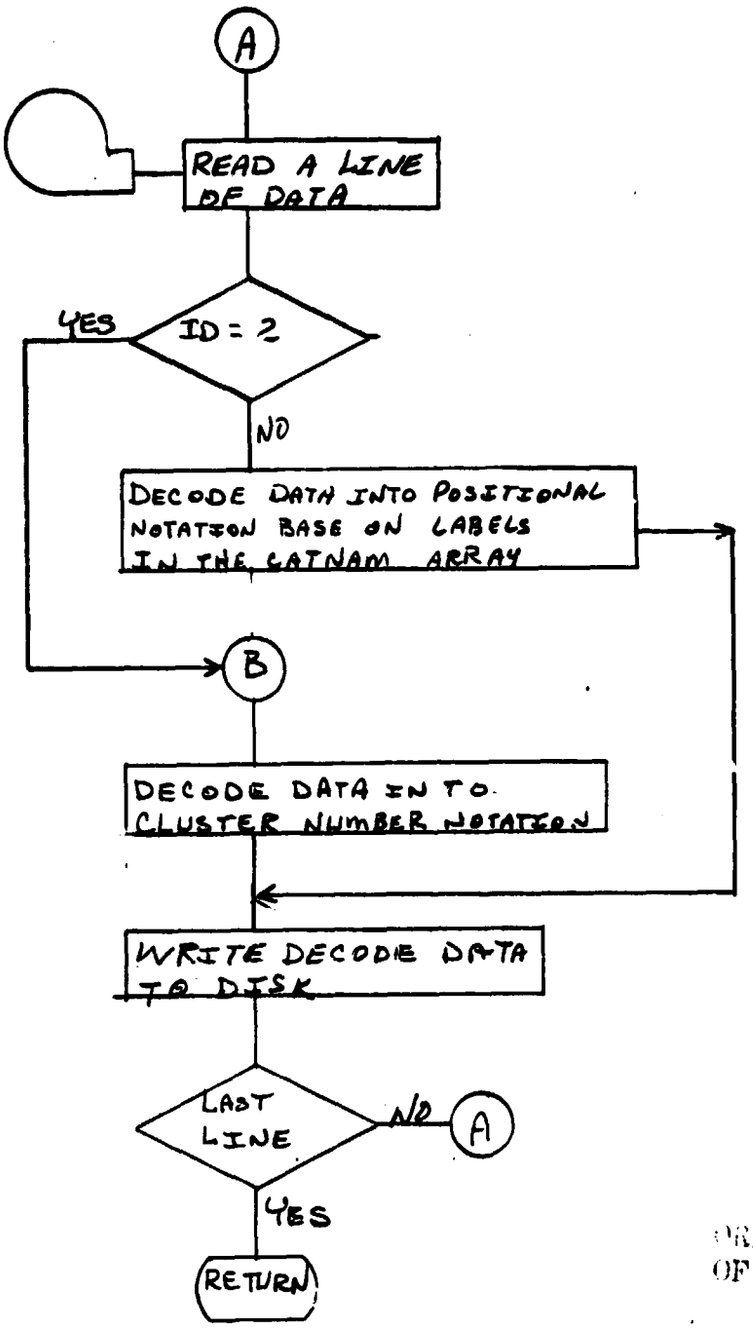


5.2 SUBROUTINE MAPUPD

```
FØRTRAN IV-PLUS V02-04          10:19:48    01-AUG-77          PAGE 6
DTERM.FTN       /TR:BLØCKS/WR
0001            SUBRØUTINE MAPUPD(ID)
0002            IMPLICIT INTEGER(A-Z)
0003            INCLUDE 'SYS[300,3]CAMSPARAM.INC'
0004 *        PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     *       1,MAXV=11,NDØTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *       2NØSPWD=6,NØDTWD=10
      C
      C
      C
0005            LØGICAL*1 TREC(144,MAXSUB),DATA(NPIX)
0006            LØGICAL*1 CØM1(938),DØTCAT(NDØTS),DØTCLU(NDØTS)
0007            EQUIVALENCE (CØM1(521),DØTCAT),(CØM1(730),DØTCLU)
0008            DATA LUN/10/
      C
0009            K=0
0010            READ(4) CØM1    ! READ INTØ CØM1 FRØM TSTAT
0011            KNT=0
0012            DØ 5 I=1,MAXSUB
0013            READ(4,END=8) (TREC(J,I),J=1,144)
0014            KNT=KNT+1
0015    5       CØNTINUE
0016    8       DØ 150 DL =1,NLIN
0017            CALL LINERD(DATA,DL)
0018            GØ TØ (10,40),ID
0019    10      CALL CLASS(NPIX,DATA)
0020            WRITE (9'DL) DATA     !UPDATE TCLAS.DAT
0021            IF(IMØD(DL,DLSKIP) .NE. 0) GØ TØ 150
0022            DØ 30 J=DSSKIP,NPIX,DSSKIP
0023            K=K+1
0024            DØTCAT(K)=DATA(J)
0025            IF(DØTCAT(K) .EQ. 0) DØTCAT(K)=-3
0026    30      CØNTINUE
0027            GØ TØ 150
0028    40      CALL CLUSTR(NPIX,DATA)
0029            WRITE (8'DL) DATA     !UPDATE TCLUS.DAT
0030            IF(IMØD(DL,DLSKIP) .NE. 0) GØ TØ 150
0031            DØ 80 J=DSSKIP,NPIX,DSSKIP
0032            K=K+1
0033            DØTCLU(K)=DATA(J)
0034    80      CØNTINUE
0035    150     CØNTINUE
0036            REWIND 4
0037            WRITE (4) CØM1
0038            DØ 160 I=1,KNT
0039            WRITE(4) (TREC(J,I),J=1,144)
0040    160     CØNTINUE
0041            END FILE 4
0042            RETURN
0043            END
```

5.2 SUBROUTINE MAPUPD

ID - IDENIFICATION OF DATA
1= CLASSIFICATION DATA
2= CLUSTER DATA



ORIGINAL PAGE IS
OF POOR QUALITY

5.3 SUBROUTINE CLASS

```
FØRTRAN IV-PLUS V02-04          10119157     01-AUG-77          PAGE 8
DTERM.FTN       /TR:BL0CKS/WR
0001                SUBR0UTINE CLASS(NPIX,DATA)
0002                IMPLICIT INTEGER(A-Z)
0003                C0MM0N /CATNAM/CATNAM(61)
0004                L0GICAL*1 DATA(NPIX)
0005                DIMENSI0N P0S(4),LABEL(4)
0006                DATA LABEL/2HW ,2HS ,2HX ,2HN /
0007                DATA P0S/239,207,79,143/
                C
                C
0008                LIMIT=0
0009                D0 150 L=1,NPIX
0010                TEMP=DATA(L)
0011                TEMP=IAND(TEMP,"377)
0012                IF(TEMP .GT. 0) G0 T0 15
0013                DATA(L)=0
0014                G0 T0 150
0015    15          D0 40   J=1,4
0016                IF(TEMP .NE. P0S(J)) G0 T0 40
0017                LAB=LABEL(J)
0018                G0 T0 80
0019    40          C0NTINUE
0020                DATA(L)=0
0021                G0 T0 150
0022    80          D0 100 J=1,60
0023                IF(LAB .NE. CATNAM(J))G0 T0 100
0024                DATA(L)=J
0025                G0 T0 150
0026    100         C0NTINUE
0027                DATA(L)=0
0028                PRINT 110,LAB
0029    110         F0RMAT(1H0,'THE CATNAM TABLE DOES N0T C0NTAIN THE',
                *   ' F0LL0WING LABEL ',A2,/,1X,'THEREF0RE THE CLASSIFICATION',
                *   ' MAP C0NTAINS A THRESH0LD VALUE F0R THIS LABEL')
0030                LIMIT=LIMIT+1
0031                IF(LIMIT .LE. 100) G0 T0 150
0032                PRINT 120
0033    120         F0RMAT(1H0,'A LIMIT 0F 100 DIAGN0STICS ',
                *   'HAS BEEN REACHED---RUN ABORTED')
0034                STOP 'DIAGN0'
0035    150         C0NTINUE
0036                RETURN
0037                END
```

5.3 SUBROUTINE CLASS

A flow chart for this subroutine is not available.

5.4 SUBROUTINE CLUSTR

```
FORTRAN IV-PLUS V02-04          10:20:04    01-AUG-77          PAGE 10
DTERM.FTN       /TR:BLOCKS/WR
0001            SUBROUTINE CLUSTR(NPIX,DATA)
0002            IMPLICIT INTEGER(A-Z)
0003            COMMON LOOKUP(239)
0004            LOGICAL*1 DATA(NPIX),LOOKUP
0005            DATA COUNT,TIMES/0,0/
        C
        C
0006            DO 50 L=1,NPIX
0007            TEMP1=DATA(L)
0008            TEMP1=IAND(TEMP1,"377)
0009            IF(TEMP1 .GT. 0 .AND. TEMP1 .LT. 240)GO TO 25
0010            DATA(L)=0
0011            GO TO 50
0012    25      DATA(L)=LOOKUP(TEMP1)+1
0013            IF(DATA(L) .LT. 0) DATA(L)=0
0014            IF(TEMP1 .EQ. 15)GO TO 50
0015            IF(DATA(L) .EQ.0) COUNT=COUNT+1
0016    50      CONTINUE
0017            TIMES=TIMES+1
0018            IF(TIMES .LT. 117)RETURN
0019            IF(COUNT .NE. 0)PRINT 100,COUNT
0020            COUNT=0
0021    100     FORMAT(1H0,'*WARNING* THERE WERE',I4,' CODES NOT RECOGNIZED')
0022            TIMES=0
0023            RETURN
0024            END
```

5.4 SUBROUTINE CLUSTR

A flow chart for this subroutine is not available.

C-3

5.5 SUBROUTINE FMAINT

```
FØRTRAN IV-PLUS V02-04          10:20:09    01-AUG-77          PAGE 12
DTERM.FTN       /TR:BLØCKS/WR
0001            SUBRØUTINE FMAINT(SEGNUM,NPIX4,NLIN,ERRØR)
0002            IMPLICIT INTEGER(A-Z)
0003            DIMENSIØN TCLUNM(15),TCLANM(15),SEGNAM(2),TSTAT(15)
0004            DATA TSTAT/2HDB,2H2:,2HC3,2H00,2H,3,2H00,2HJX,2HXX,2HXT,2HST,
               *          2HAT,2H,D,2HAT,2H;1,0/
0005            DATA TCLUNM/2HDB,2H2:,2HC3,2H00,2H,3,2H00,2HJX,2HXX,2HXT,2HCL,
               *          2HIS,2H,M,2HAP,2H;1,0/
0006            DATA TCLANM/2HDB,2H2:,2HC3,2H00,2H,3,2H00,2HJX,2HXX,2HXT,2HCL,
               *          2HAS,2H,M,2HAP,2H;1,0/
          C
0007            ERRØR=0
          C
0008            CALL GNAME(SEGNUM,SEGNAM)
0009            CALL SUBSTR(SEGNAM,1,4,TCLUNM,14,4)
0010            CALL SUBSTR(SEGNAM,1,4,TCLANM,14,4)
0011            CALL SUBSTR(SEGNAM,1,4,TSTAT,14,4)
0012            ØPEN(UNIT=4,NAME=TSTAT,TYPE='ØLD',ACCESS='SEQUENTIAL',
               *     ERR=150,FØRM='UNFØRMATTED')
0013            ØPEN(UNIT=8,NAME=TCLUNM,TYPE='UNKNØWN',ACCESS='DIRECT',
               * RECØRDSIZE=NPIX4,ERR=150,MAXREC=NLIN,ASSØCIATEVARIABLE=V2)
0014            ØPEN(UNIT=9,NAME=TCLANM,TYPE='UNKNØWN',ACCESS='DIRECT',
               * RECØRDSIZE=NPIX4,ERR=150,MAXREC=NLIN,ASSØCIATEVARIABLE=V3)
0015            RETURN
0016            ENTRY FADE
0017            CLØSE(UNIT=4,DISPØSE='SAVE')
0018            CLØSE(UNIT=8,DISPØSE='SAVE')
0019            CLØSE(UNIT=9,DISPØSE='SAVE')
0020            RETURN
0021    150     ERRØR=1
0022            RETURN
0023            END
```

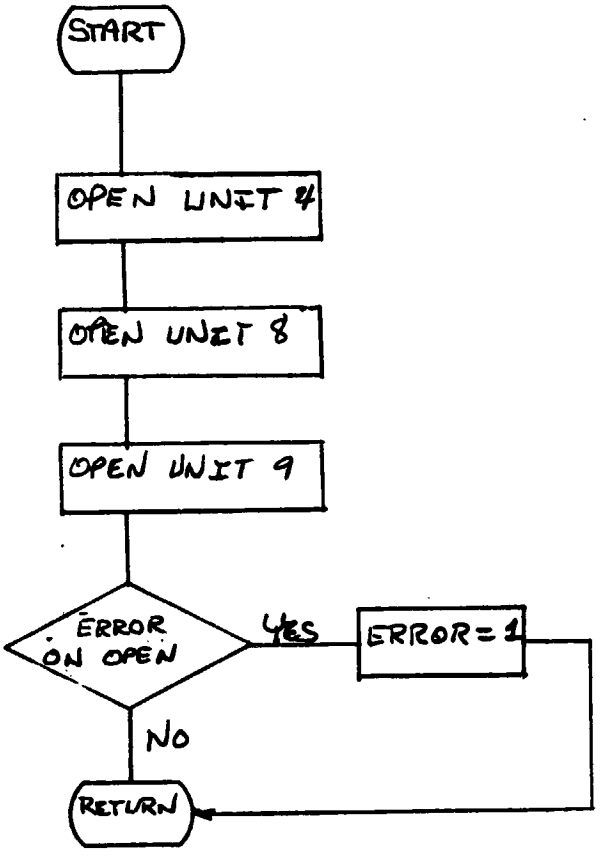
ORIGINAL PAGE IS
OF POOR QUALITY

5.5 SUBROUTINE FMAINT

SEGNUM - SEGMENT NUMBER

!PIX4 - NUMBER OF DOUBLE WORDS IN DIRECT ACCESS FILE

NLIN - NUMBER OF LINES



5.6 SUBROUTINE SORTRC

```
FØRTRAN IV-PLUS V02-04          10:20:14     01-AUG-77          PAGE 14
DTERM.FTN       /TR:BLØCKS/WR
0001            SUBRØUTINE SØRTRC(A,ITEMS)
0002            IMPLICIT INTEGER (A-Z)
0003            DIMENSIØN TEMP(2),A(2,ITEMS)
0004            M=ITEMS
0005    1       M=M/2
0006            IF(M .EQ. 0)GØ TØ 10
0007            K=ITEMS-M
0008            J=1
0009    2       I=J
0010    3       IM=I+M
0011            IF(A(2,I)-A(2,IM)) 4,4,5
0012    5       CALL SUBSTR(A(1,I),1,4,TEMP,1,4)
0013            CALL SUBSTR(A(1,IM),1,4,A(1,I),1,4)
0014            CALL SUBSTR(TEMP,1,4,A(1,IM),1,4)
0015            I=I-M
0016            IF(I .GE. 1) GØ TØ 3
0017    4       J=J+1
0018            IF(J .GT. K) GØ TØ 1
0019            GØ TØ 2
0020    10      RETURN
0021            END
```

5.6 SUBROUTINE SORTRC

A flow chart for this subroutine is not available.

5.7 SUBROUTINE OPMESS

```
FØRTRAN IV-PLUS V02-04          10:20:18    01-AUG-77          PAGE 16
DTERM.FTN       /TR:BLØCKS/WR
0001            SUBRØUTINE ØPMESS(DISKNM)
0002            INTEGER DISKNM
0003            LØGICAL*1 ANS(16),BLANK
0004            DATA BLANK /1H /
          C
          C
          C
0005      20    TYPE 30,DISKNM,DISKNM
0006      30    FØRMAT(1HØ,'THE NEXT SEGMENTS TØ BE PRØCESSED WILL BE FØUND ',
                *            ' DISK NUMBER ',I2,//,1X,'IF YØU WISH TØ CØNTINUE',
                *            ' PLEASE MØUNT DISK NUMBER ',I2,/,1X,
                *            'THEN TYPE THE LETTER (C) AND CARRIAGE RETURN.',/,
                *      1HØ,'IF YØU WISH TØ ABØRT THIS JØB THEN TYPE IN THE ',
                *            'LETTER (X) AND CARRIAGE RETURN',10X,
                *            'THANK YØU')
          C
          C
0007            ACCEPT 40,ANS
0008      40    FØRMAT(16A1)
0009            DØ 50 J=1,16
0010            IF(ANS(J) .NE. BLANK) GØ TØ 60
0011      50    CØNTINUE
0012            GØ TØ 20
0013      60    IF(ANS(J) .EQ. 'Y') RETURN
0014            TYPE 70
0015      70    FØRMAT(1HØ,'ANALYST ABØRTED RUN')
0016            STØP 'ABØRT'
0017            END
```

5.7 SUBROUTINE OPMESS

A flow chart for this subroutine is not available.

5.8 SUBROUTINE JULIAN

```
FØRTRAN IV-PLUS V02-04          10120122     01-AUG-77          PAGE 16
DTERM.FTN                       /TR:BLØCKS/WR
0001                FUNCTIØN JULIAN(MØ,DAY,YR)
0002                INTEGER TØTAL,DAY,YR
0003                DIMENSIØN MDAYS(12)
0004                DATA MDAYS/31,0,31,30,31,30,31,31,30,31,30,31/
0005                MDAYS(2)=28
0006                IF(IMØD(YR,4) .EQ. 0) MDAYS(2)=29
0007                TØTAL=0
0008                LEN=MØ-1
0009                IF(LEN)10,10,20
0010      10        JULIAN=DAY
0011                RETURN
0012      20        DØ 30 I=1,LEN
0013                TØTAL=TØTAL+MDAYS(I)
0014      30        CØNTINUE
0015                JULIAN=TØTAL+DAY
0016                RETURN
0017                END
```

5.8 SUBROUTINE JULIAN

A flow chart for this subroutine is not available.

5.9 SUBROUTINE HEADIN

```
FORTRAN IV-PLUS V02-04          10:20:25    01-AUG-77          PAGE 20
DTERM.FTN       /TR:BLOCKS/WR
0001              SUBROUTINE HEADIN(ID,SEGNUM,JULDAT,YR,EOF)
0002              IMPLICIT INTEGER(A-Z)
0003              INCLUDE 'SYI:[300,30]CAMSPARAM.INC'
0004    *       PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
        *      1,MAXV=11,NDOTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
        *      2NOSPWDS=6,NODTWD=10
0005              COMMON /WCOM/ SS,SE,LS,LE,NRPDS,NDSPR,NCPR,NRPC,ANCL,NC,NS,NBIT,
                 * DOI,NCAR,SVD,RSIZ,PSKIP,HSIZ,CALP,CERR
0006              COMMON /IOLP/ IO,LP        !THESE TWO COMMONS ARE TED KELLY'S
0007              LOGICAL*1 BUFLOG(3072),DATA(*PIX),FILNAM(33),HEADER(3060),TEMP
0008              DIMENSION HEADWD(1530)
0009              EQUIVALENCE(HEADER,HEADWD)
0010              DATA LUN/10/,F11/1/,FORMAT/1/,SHFG/1/,FN/0/,RWD/0/
0011              DATA MTXTFG/1/,UNIT/1/
                C     F11=1      ! 0=FILES 11 ; 1=FOREIGN
                C
                C
                C
                C     FORMAT=1    ! 1=UNIVERSAL FORMAT
                C     SHFG=1      ! 1=DO NOT SAVE HEADER
                C
                C
                C
                C
0012                LP=6
0013              IO=5
0014              IF(ID .EQ. -9)FN=0
0015    20        CALL DSET(LUN,F11,FILE,MTXTFG,UNIT,FN,RWD)
0016              CALL HREAD(LUN,HEADER,BUFLOG,3072,FORMAT,F11,PRTY,EOF,SHFG)
0017              FN=1
0018              RWD=2
0019              IF(CERR .NE. 0) STOP 'CERR'
0020              IF(EOF .EQ. 0) RETURN
0021              TEMP=HEADER(67)
0022              HEADER(67)=HEADER(68)
0023              HEADER(68)=TEMP
0024              SEGNUM=HEADWD(34)
0025              ID=HEADER(53)
0026              ID=IAND(ID,"377)
0027              ID=ID-240
                C
                C   ID=1; CLASSIFICATION MAP
                C     =2; REGULAR CLUSTER MAP
                C     =3; CONDITIONAL MAP
                C
0028              IF(ID .EQ. 3) GO TO 20
0029              MO=HEADER(62)
0030              DAY=HEADER(61)
0031              YR=HEADER(63)
0032              JULDAT=JULIAN(MO,DAY,YR)
0033              CALL FFIND(LUN,1,1,NPIX,NLIN,FORMAT,BUFLOG,3072)
0034              RETURN
                C
0035              ENTRY LINERD(DATA,DL)
0036              PRTY=0
0037              CALL LREED(LUN,DATA,BUFLOG,3072,DL,F11,PRTY,EOF,FORMAT,1)
0038              IF(EOF .NE. 0) RETURN
```

```
0039            TYPE 200
0040    200     FORMAT(1HO,'ERROR---ENCOUNTERED EOF WHILE READING',
               *  ' DATA FROM TAPE',/,1X,'UNRECOVERABLE CONDITION',
               *  '---RUN ABORTED')
0041            STOP 'LINERR'
0042            END
```

ORIGINAL PAGE IS
OF POOR QUALITY

5.9 SUBROUTINE HEADIN

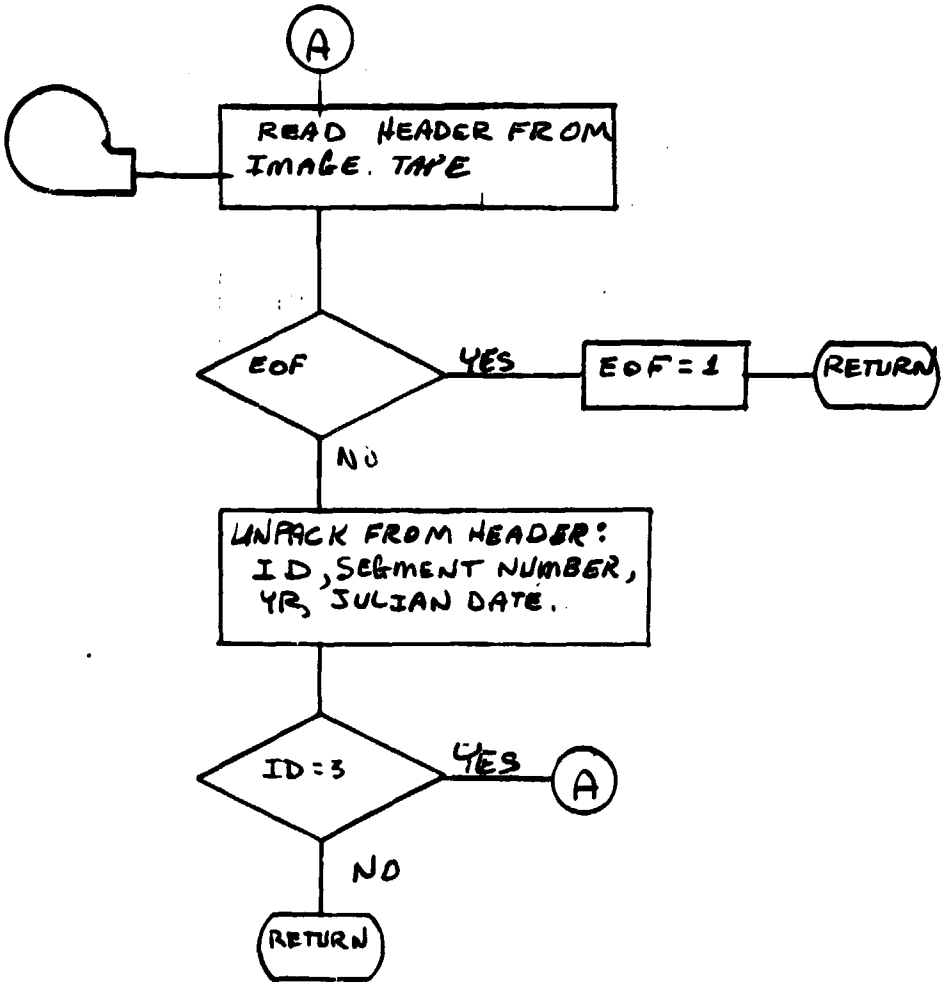
ID = : IDENIFICATION OF DATA
1 = CLASSIFICATION
2 = UNCONDITIONAL CLUSTER
3 = CONDITIONAL CLUSTER

SEGNUM - SEGMENT NUMBER

JULDAT - JULIAN DATE

YR - YEAR

EOF - END OF FILE



5.10 SUBROUTINE REPORT

```
FORTRAN IV-PLUS V02-04          10120132     01-AUG-77          PAGE 24
DTERM.FTN       /TR:BLOCKS/WR
0001            SUBROUTINE REPORT(AID,ASEG,ADSK,AYR,AJUL,FILNUM,T1)
0002            IMPLICIT INTEGER(A-Z)
0003            DIMENSION AID(30),ASEG(30),ADSK(30),AYR(30),AJUL(30)
0004            LOGICAL*1 IDATE(9),ITIME(8)
                C
                C
                C
0005            CALL DATE(IDATE)
0006            CALL TIME(ITIME)
0007            PRINT 5
0008      5     FORMAT(1H0,35X,'CAMS I-100 DATA BASE TRANSACTION REPORT')
0009            PRINT 10,IDATE,ITIME
0010      10    FORMAT(1H0,100X,'DATE ',9A1,/,101X,'TIME ',8A1)
0011            PRINT 15
0012      15    FORMAT(1H0,35X,'ERIPS DTERM INTERFACE UPDATE',' / MAY 77')
0013            PRINT 17
0014      17    FORMAT(1H0,46X,'DISK',3X,'DATE OF ERIPS')
0015            PRINT 18
0016      18    FORMAT(35X,'SEGMENT',6X,'ID',6X,
                *  'PROCESSING',8X,'MAP TYPE')
                C
                C
                C
0017            DO 100 J=1,FILNUM
0018            GO TO (50,70),AID(J)
0019      50    PRINT 60,ASEG(J),ADSK(J),AYR(J),AJUL(J)
0020      60    FORMAT(36X,I4,8X,I2,8X,I2,1X,I3,8X,'CLASSIFICATION')
0021            GO TO 100
0022      70    PRINT 85,ASEG(J),ADSK(J),AYR(J),AJUL(J)
0023      85    FORMAT(36X,I4,8X,I2,8X,I2,1X,I3,8X,'CLUSTER')
0024      100   CONTINUE
                C
                C
                C
0025            CALL DUSET(10,1,0)
0026            PRINT 110
0027      110   FORMAT(1H1)
0028            STOP 'ALL OK'
0029            END
```

5.10 SUBROUTINE REPORT

A flow chart for this subroutine is not available.

5.11 DATA LOOKUP

```
FORTRAN IV-PLUS V02-04          10:20:36    01-AUG-77          PAGE 26
DTERM.FTN       /TR:BLOCKS/NR
0001            BLOCK DATA
0002            COMMON L00KUP(239)
0003            LOGICAL*1 L00KUP
0004            DATA L00KUP/1,32*0,30,0,23,0,0,9,0,58,44,16,0,51,37,2,18*0,31,0,
                *           24,0,45,10,0,59,0,17,0,52,38,3,18*0,32,0,25,0,46,11,
                *           0,60,39,18,0,53,0,4,20*0,26,0,47,12,0,61,40,19,0,54,
                *           33,5,20*0,27,0,48,13,0,0,41,20,0,55,34,6,20*0,28,0,49,
                *          14,0,0,42,21,0,56,35,7,20*0,29,0,50,15,0,57,43,22,0,0,
                *          36,8/
0005            END
```

5.11 DATA LOOKUP

A flow chart for this subroutine is not available.

5.12 SUBROUTINE ZNAME

```
FØRTRAN IV-PLUS V02-04          10:20:37    01-AUG-77          PAGE 28
DTERM.FTN       /TR:BLØCKS/WR

0001            SUBRØUTINE ZNAME(SEGNUM,SEGNAM)
0002            IMPLICIT INTEGER(A-Z)
0003            DIMENSIØN SEGNAM(2),NAME(2)
0004            LØGICAL*1 CHAR(4)
0005            EQUIVALENCE (NAME,CHAR)
0006            ENCØDE(4,20,CHAR) SEGNUM
0007    20      FØRMAT(I4)
0008            IF(SEGNUM .GE. 1000) GØ TØ 60
0009            DØ 50 J=1,3
0010            IF(CHAR(J) .NE. ' ' ) GØ TØ 60
0011            CHAR(J)='0'
0012    50      CØNTINUE
0013    60      SEGNAM(1)=NAME(1)
0014            SEGNAM(2)=NAME(2)
0015            RETURN
0016            END
```

5.12 SUBROUTINE ZNAME

A flow chart for this subroutine is not available.

6. SEGMENT DELETE PROGRAM SEGDEL

```
HFØRTRAN IV-PLUS V02-04                  10131120    01-JUL-77          PAGE 1
SEGDEL.FTN      /TR:BLØCKS/WR
      C*****************************************************************
      C                                                                *
      C                          SEGMENT DELETE                        *
      C                                                                *
      C*****************************************************************
0001          IMPLICIT INTEGER (A-W)
0002          INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0003  *       PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      *      1,MAXV=11,ND0TS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *      2N0SPWD=6,N0DTWD=10
0004          DIMENSI0N A(4),SEGN0(2),FILNAM(15),FILTYP(25),KPAT01(2,200)
0005          DIMENSI0N BUFF2(256),BUFF3(2)
0006          L0GICAL*1 NAMFIL(30),BUFF1(512),KEN(30)
0007          EQUIVALENCE (FILNAM(1),NAMFIL(1))
0008          EQUIVALENCE (BUFF1(1),BUFF2(1))
0009          REAL X1,X2,X3
0010          DATA FILTYP/2HDA,2HTM,2HAP,2HPC,2HLA,2HST,2HCL,2HAS,2HPC,2HLU,
             *  2HST,2HCL,2HUS,2HPS,2HTA,2HTT,2HST,2HAT,2HDD,2H0T,2HSF,2HIE,
             *  2HLD,2H00,2HDK/
0011          DATA FILNAM/2HDB,2H2:,2H[3,2H03,2H,3,2H00,2H] ,2H  ,2H  ,2H  ,
             *  2H  ,2H.D,2HAT,2H;1,0/
0012       100 F0RMAT(1H1,40X,'CAMS I-100 DATA BASE TRANSACTI0N REP0RT')
0013       101 F0RMAT(1H0,103X,'DATE ',I2,'/',I2,'/',I2)
0014       102 F0RMAT(1H0,103X,'TIME ',4A2)
0015       103 F0RMAT(1H0,4X,'SEGMENT NUMBER   = ',2A2)
0016       104 F0RMAT(1H0,4X,'DISK PACK NUMBER = ',I4)
0017       105 F0RMAT(1H0,'WR0NG DISK PACK M0UNTED')
0018       106 F0RMAT(1HS,'INPUT SEGMENT NUMBER  >')
0019       107 F0RMAT(2A2)
0020       108 F0RMAT(1H0,'SEGMENT NUMBER N0T 0N DATA BASE')
0021       109 F0RMAT(I2)
0022       110 F0RMAT(2I4)
0023       111 F0RMAT(1H0,4X,'T0TAL ELAPSED TIME = ',I2,':',I2,':',I2)
0024       112 F0RMAT(1H0,44X,'SEGMENT DELETE/VERSI0N MAY 31,1977')
0025       113 F0RMAT(1H0,47X,'FILE',9X,'FILE',4X,'FILE N0T')
0026       114 F0RMAT(1H0,47X,'NAME',8X,'DELETED',2X,'DELETED')
0027       115 F0RMAT(1H0,44X,13A1,5X,'X')
0028       116 F0RMAT(1H0,44X,13A1,14X,'X')
0029       117 F0RMAT(1H0,1X,I4,I2,4I3,2I3,1X,10A1,1X,17I3)
0030       118 F0RMAT (I3)
0031       119 F0RMAT (I4)
0032       120   F0RMAT(1HS,'KEYIN T0 (D)ELETE AN0THER SEGMENT  >')
0033       121   F0RMAT(30A1)
0034       122   F0RMAT(1H0,'PLEASE M0UNT DISK PACK N0, ',I4)
0035       123   F0RMAT(1HS,'D0 YOU WISH T0 (C)0NTINUE?  >')
0036          CALL ASSIGN(6,'LP:')
0037            CALL ERRSET(29,,TRUE.,,FALSE.,,TRUE.,,FALSE.,,80)
0038            II=1
0039            CALL ELAPSE(II,IHR,IMIN,ISEC)
0040          X1=SECNDS(0.)
0041        1 TYPE 106
0042          ACCEPT 107,SEGN0
0043          DEC0DE(4,119,SEGN0) SEGNA
0044          CALL DSKCHK(SEGNA,PTR,DSKID,FLAG)
0045          IF(FLAG .EQ. 1) G0 T0 301
0046          IF(FLAG .EQ. 2) G0 T0 302
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
SEGDEL.FTN      /TR:BLØCKS/WR
0047            CALL IDATE(I,J,K)
0048            CALL TIME(A)
0049            WRITE(6,100)
0050            WRITE(6,101) I,J,K
0051            WRITE(6,102) (A(P),P=1,4)
0052            WRITE(6,103) (SEGN3(P),P=1,2)
0053            WRITE(6,104) DSKID
0054            WRITE(6,112)
0055            WRITE (6,113)
0056            WRITE (6,114)
0057            IF(DSKID .GE. 1000)  CALL SUBSTR(FILTYP,49,2,FILNAM,1,2)
0058            RECTYP=1
0059            L=3
0060            ØPEN(UNIT=3,NAME='DB0:[300,300]DSKTBL.DAT',ACCESS='DIRECT',
               *  TYPE='ØLD',MAXREC=1,RECØRDSIZE=200)
0061            ØPEN(UNIT=4,NAME='SY0:[300,300]DIRFILE.DAT',TYPE='ØLD',
               *  MAXREC=200,ACCESS='DIRECT',RECØRDSIZE=128)
0062            FIND (4'PTR)
0063            READ (4'PTR) BUFF2
0064            NØACQ=BUFF1(4)
0065              CALL SUBSTR(FILTYP,1,3,FILNAM,24,3)
0066            CALL SUBSTR(SEGNØ,1,4,FILNAM,14,4)
0067         5  GØ TØ (10,20,30,40,50,60,70,80,90,500),RECTYP
        C         SEGNØ=DDØ:S.DAT
0068        10  CALL SUBSTR(FILTYP,37,5,FILNAM,18,5)
0069            ØPEN(UNIT=1,NAME=FILNAM,TYPE='ØLD',MAXREC=NDØTS,
               * ACCESS='DIRECT',RECØRDSIZE=13,ERR=205)
0070            GØ TØ 200
        C         SEGNØ=FIELD.DAT
0071        20  CALL SUBSTR(FILTYP,42,5,FILNAM,18,5)
0072            ØPEN(UNIT=1,NAME=FILNAM,TYPE='ØLD',MAXREC=MAXFLD,
               * ACCESS='SEQUENTIAL',FØRM='UNFØRMATTED',ERR=205)
0073            GØ TØ 200
        C         SEGNØ=PSTAT.DAT
0074        30  CALL SUBSTR(FILTYP,27,5,FILNAM,18,5)
0075            ØPEN(UNIT=1,NAME=FILNAM,TYPE='ØLD',MAXREC=MAXCAT,
               * ACCESS='SEQUENTIAL',FØRM='UNFØRMATTED',ERR=205)
0076            GØ TØ 200
        C         SEGNØ=TSTAT.DAT
0077        40  CALL SUBSTR(FILTYP,32,5,FILNAM,18,5)
0078            ØPEN(UNIT=1,NAME=FILNAM,TYPE='ØLD',MAXREC=MAXCAT,
               * ACCESS='SEQUENTIAL',FØRM='UNFØRMATTED',ERR=205)
0079            GØ TØ 200
        C         SEGNØ=IMAGERY.DAT
0080        50  DØ 51 K=1,NØACQ
0081            ENCØDE(2,109,BUFF3) BUFF2(L)
0082            CALL SUBSTR(BUFF3,1,2,FILNAM,18,2)
0083            ENCØDE(3,118,BUFF3) BUFF2(L+1)
0084            CALL SUBSTR(BUFF3,1,3,FILNAM,20,3)
0085            IF(BUFF2(L+1) .LE. 9)   CALL SUBSTR(FILTYP,49,2,FILNAM,20,2)
0086            IF(BUFF2(L+1) .LE. 99)  CALL SUBSTR(FILTYP,49,1,FILNAM,20,1)
0087            L=L+2
0088            ØPEN(UNIT=1,NAME=FILNAM,TYPE='ØLD',MAXREC=NLIN,
               * ACCESS='SEQUENTIAL',FØRM='UNFØRMATTED',ERR=205)
0089            WRITE(6,115)(NAMFIL(P),P=14,26)
0090            CLØSE (UNIT=1,DISPØSE='DELETE',ERR=205)
```

```
0091        51 CØNTINUE
0092           GØ TØ 201
            C     SEGNØ=PCLAS,MAP
0093        60 CALL SUBSTR(FILTYP,4,3,FILNAM,24,3)
0094           CALL SUBSTR(FILTYP,7,5,FILNAM,18,5)
0095           ØPEN(UNIT=1,NAME=FILNAM,TYPE='ØLD',MAXREC=NLIN,
              * ACCESS='DIRECT',RECØRDSIZE=49,ERR=205)
0096           GØ TØ 200
            C     SEGNØ=TCLAS,MAP
0097        70 CALL SUBSTR(FILTYP,12,5,FILNAM,18,5)
0098           ØPEN(UNIT=1,NAME=FILNAM,TYPE='ØLD',MAXREC=NLIN,
              * ACCESS='DIRECT',RECØRDSIZE=49,ERR=205)
0099           GØ TØ 200
            C     SEGNØ=PCLUS,MAP
0100        80 CALL SUBSTR(FILTYP,17,5,FILNAM,18,5)
0101           ØPEN(UNIT=1,NAME=FILNAM,TYPE='ØLD',MAXREC=NLIN,
              * ACCESS='DIRECT',RECØRDSIZE=49,ERR=205)
0102           GØ TØ 200
            C     SEGNØ=TCLUS,MAP
0103        90 CALL SUBSTR(FILTYP,22,5,FILNAM,18,5)
0104           ØPEN(UNIT=1,NAME=FILNAM,TYPE='ØLD',MAXREC=NLIN,
              * ACCESS='DIRECT',RECØRDSIZE=49,ERR=205)
0105           GØ TØ 200
0106       500   READ(3'1)KPAT01
0107             DØ 502 J=1,200
0108             IF(SEGNA .EQ. KPAT01(1,J)) GØ TØ 503
0109       502   CØNTINUE
0110       503   KPAT01(1,J)= -9
0111             KPAT01(2,J)=0
0112             WRITE(3'1) KPAT01
0113           DØ 501 B=1,256
0114           BUFF2(B)=0
0115       501 CØNTINUE
0116           WRITE(4'PTR) BUFF2
0117           CLØSE(UNIT=4,DISPØSE='SAVE')
0118           CLØSE(UNIT=3,DISPØSE='SAVE')
0119             II=2
0120             CALL ELAPSE(II,IHR,IMIN,ISEC)
0121           TYPE 111,IHR,IMIN,ISEC
0122           WRITE(6,111) IHR,IMIN,ISEC
0123             TYPE 120
0124             ACCEPT 121,KEN
0125             CALL FRØNT(KEN,30)
0126             IF(KEN(1) .EQ. 'D') GØ TØ 1
0127           STØP
0128       200 CLØSE (UNIT=1,DISPØSE='DELETE',ERR=205)
0129           WRITE(6,115)(NAMFIL(P),P=14,26)
0130       201 RECTYP=RECTYP+1
0131           GØ TØ 5
0132       205 CLØSE (UNIT=1,DISPØSE='DELETE')
0133           WRITE(6,116)(NAMFIL(P),P=14,26)
0134           RECTYP=RECTYP+1
0135           GØ TØ 5
0136       301 TYPE 105
0137             TYPE 122,DSKID
0138             TYPE 123
```

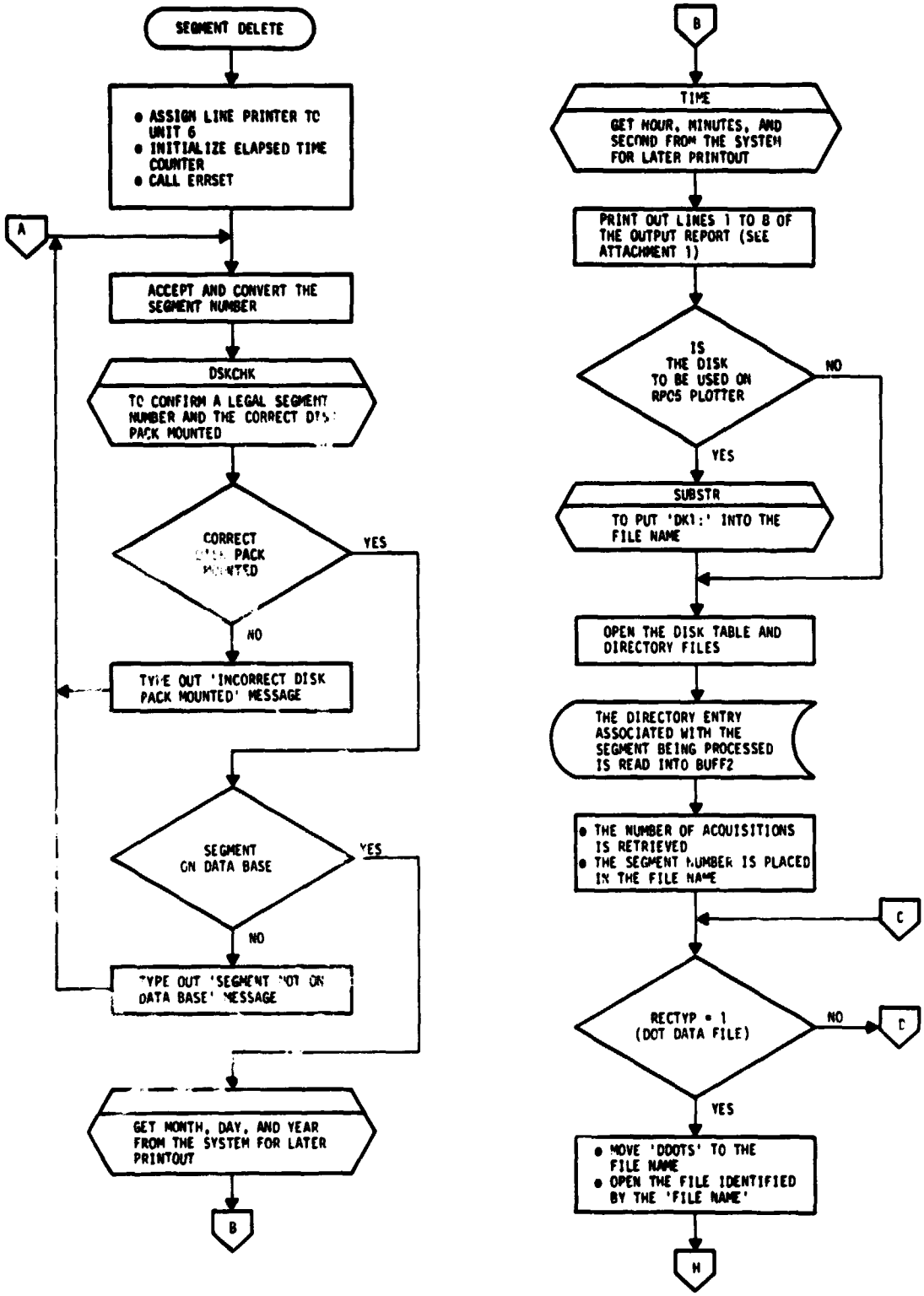
```
FØRTRAN IV+PLUS V02-04          10131120     01-JUL-77          PAGE 4
SEGDEL.FTN      /TR:BLØCKS/WR
0139            ACCEPT 121,KEN
0140            CA   FRØNT(KEN,30)                                    LI
0141            IF(KEN(1) .EQ. 'C') GØ TØ 1
0142            STØP 2
0143       302  TYPE 108
0144            GØ TØ 1
0145            END
```

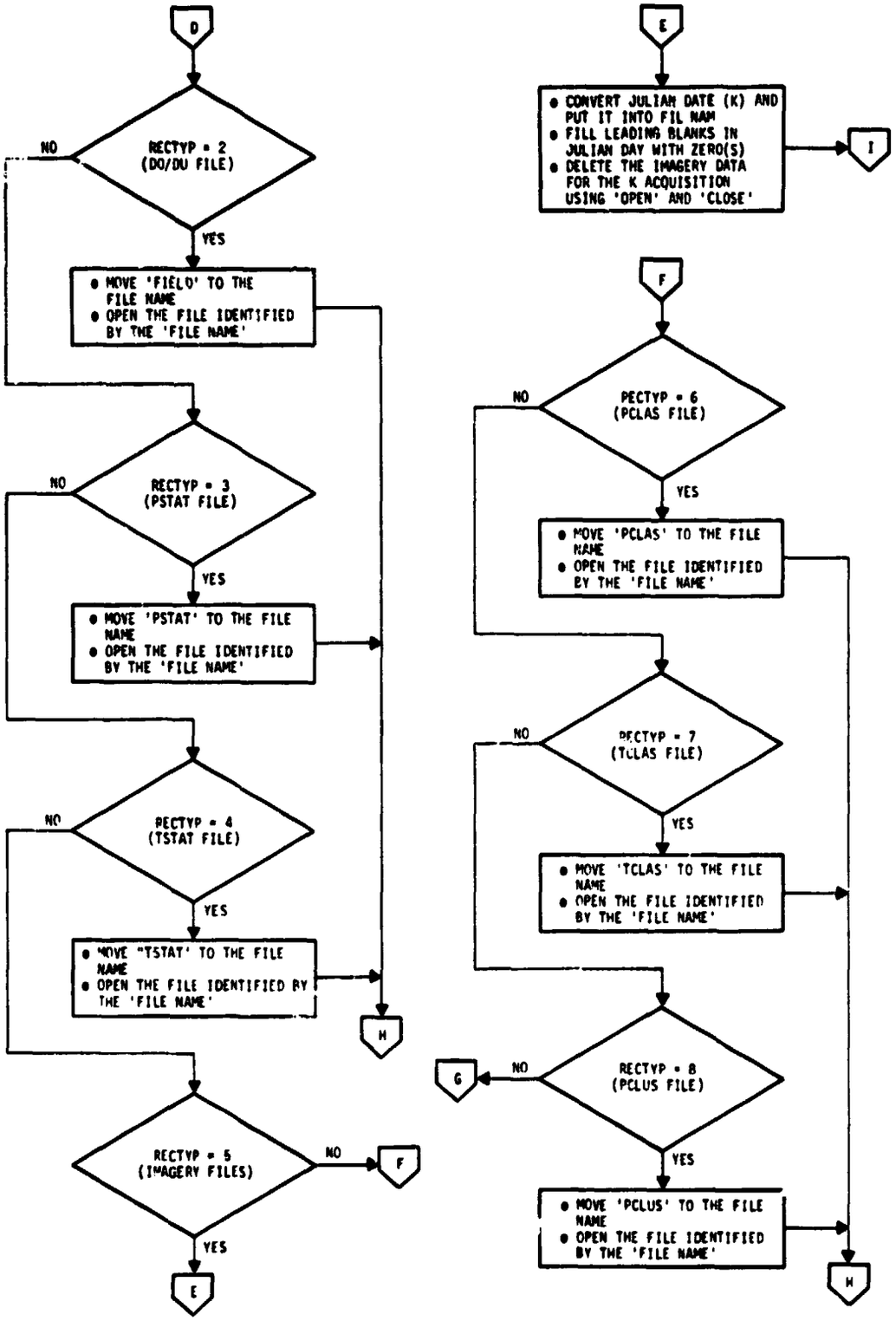
```
FØRTRAN IV+PLUS V02-04          10131147    01-JUL-77          PAGE 7
SEGDEL.FTN      /TR:BLØCKS/WR
0001            SUBRØUTINE SUBSTR(A,I,N,B,J,M)
      C MØVE A TØ B
0002            IMPLICIT INTEGER (A-Z)
0003            LØGICAL*1 A(1),B(1)
0004            DATA BLANK/2H  /
0005            IS=I
0006            JS=J
0007            L=0
0008            IF(N .EQ. 0) GØ TØ 20
0009            L=N
0010            IF(L.GT.M) L=M
0011            DØ 10 K=1,L
0012            B(JS)=A(IS)
0013            IS=IS+1
0014            JS=JS+1
0015         10 CØNTINUE
0016            IF(N.GE.M) RETURN
0017         20 L=L+1
0018            DØ 30 K=L,M
0019            B(JS)=BLANK
0020            JS=JS+1
0021         30 CØNTINUE
0022            RETURN
0023            END
```

6.1 SUBROUTINE ELAPSE

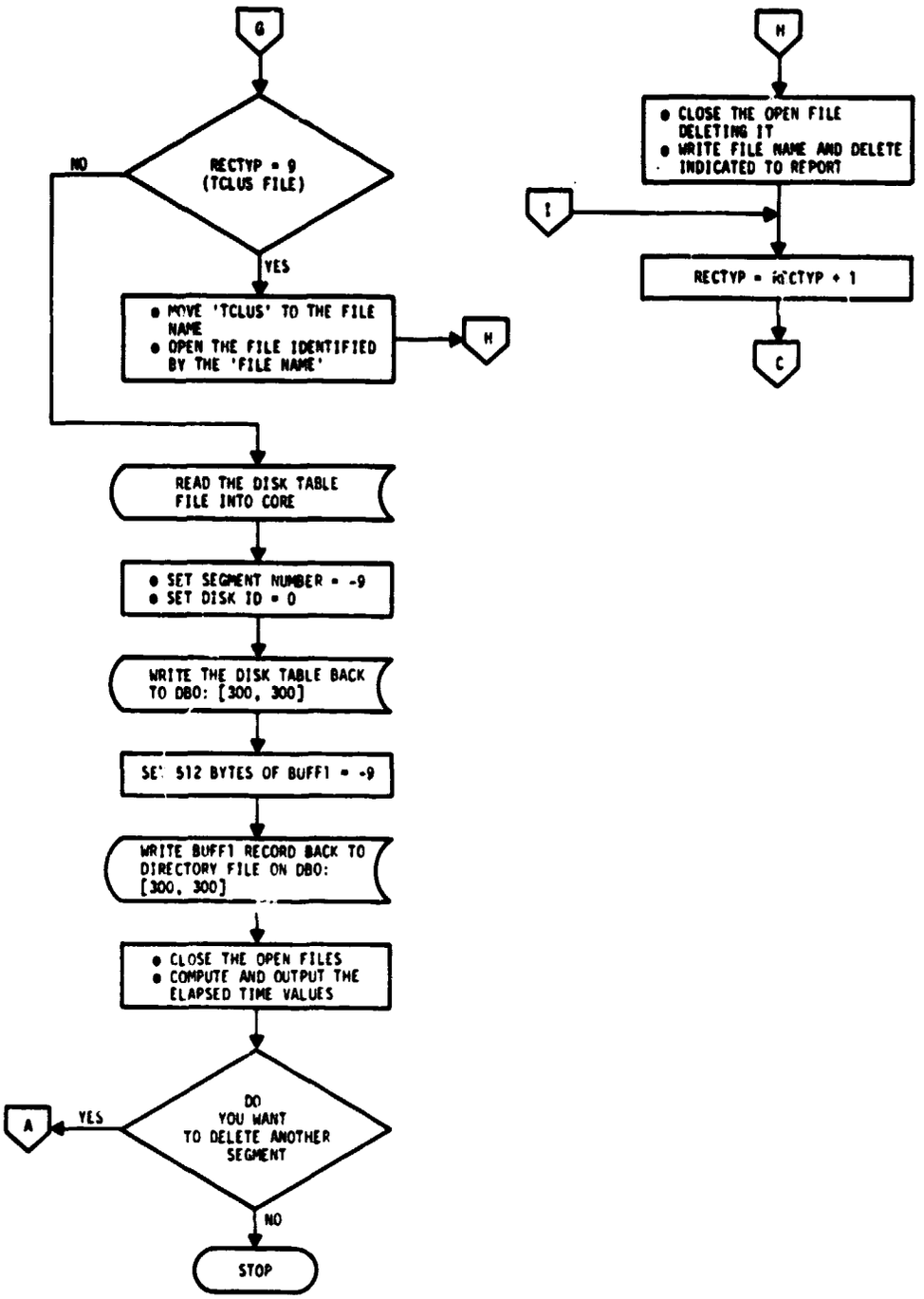
```
FØRTRAN IV-PLUS V02-04          10:31:50     01-JUL-77          PAGE 1
SEGDEL.FTN       /TR:BLØCKS/WR
0001               SUBRØUTINE ELAPSE(II,IHR,IMIN,ISEC)
0002            REAL T1,DELTA
0003            REAL SCR
0004            INTEGER JSCR,ISEC,IMIN,IHR
0005            GØ TØ (1,2),II
                C
                C       INITIAL CLALL
                C
0006          1 T1=SECNDS(0.)
0007            GØ TØ 9999
                C
                C       FINAL CALL
                C
0008          2 DELTA=SECNDS(T1)
0009            JSCR=DELTA/60.
0010            SCR=JSCR*60.
0011            ISEC=DELTA-SCR+0.5
0012            IHR=JSCR/60
0013            IMIN=JSCR-IHR*60
0014            TYPE 1000,IHR,IMIN,ISEC
0015       1000 FØRMAT(' TØTAL ELAPSED TIME= ',I2,':',I2,':',I2)
0016       9999 RETURN
0017            END
```

6. SEGMENT DELETE PROGRAM




D
RECTYP = 2
(DO/DU FILE)
NO
YES
• MOVE 'FIELD' TO THE FILE NAME
• OPEN THE FILE IDENTIFIED BY THE 'FILE NAME'
RECTYP = 3
(PSTAT FILE)
NO
YES
• MOVE 'PSTAT' TO THE FILE NAME
• OPEN THE FILE IDENTIFIED BY THE 'FILE NAME'
RECTYP = 4
(TSTAT FILE)
NO
YES
• MOVE 'TSTAT' TO THE FILE NAME
• OPEN THE FILE IDENTIFIED BY THE 'FILE NAME'
RECTYP = 5
(IMAGERY FILES)
NO
YES
E
F
H
E
• CONVERT JULIAN DATE (K) AND PUT IT INTO FIL NAM
• FILL LEADING BLANKS IN JULIAN DAY WITH ZERO(S)
• DELETE THE IMAGERY DATA FOR THE K ACQUISITION USING 'OPEN' AND 'CLOSE'
I
F
RECTYP = 6
(PCLAS FILE)
NO
YES
• MOVE 'PCLAS' TO THE FILE NAME
• OPEN THE FILE IDENTIFIED BY THE 'FILE NAME'
RECTYP = 7
(TCLAS FILE)
NO
YES
• MOVE 'TCLAS' TO THE FILE NAME
• OPEN THE FILE IDENTIFIED BY THE 'FILE NAME'
RECTYP = 8
(PCLUS FILE)
NO
YES
G
• MOVE 'PCLUS' TO THE FILE NAME
• OPEN THE FILE IDENTIFIED BY THE 'FILE NAME'
H

ORIGINAL PAGE IS
OF POOR QUALITY



7. CONTROL PROGRAM CAMSEX

```
HFORTRAN IV-PLUS V02-04          12:26:45   29-AUG-77          PAGE 1
CAMSEX.FTN       /TR:BLOCKS/WR
        C      CAMS/1-100 CONTROL PROGRAM
        C
        C      SOURCE:CAMSEX.FTN
        C      OBJECT:CAMSEX.OBJ
        C      TASK  :CAMSEX.TSK
0001            IMPLICIT INTEGER (A-O),(S-Z)
0002            INCLUDE 'SY:[300,3]CAMSCOMON.INC'
0003  *         INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0004  *         PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      *        1,MAXV=11,NDOTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *        2VOSPWD=6,NODTWD=10
0005  *         EQUIVALENCE (C1,ACDATE),(C2,ISEG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0006  *         INTEGER C1(469),C2(256),C3(71),C4(348),C5(629)
      *   C*
0007  *         INTEGER ACDATE,SUBCAT,SUBPOP,CATKNT,CATTH
0008  *         BYTE CHNVEC,NOCHAN,NOSUB,DOTCAT,DOTCLU
0009  *         COMMON/COM1/ACDATE(2,MAXACC),CHNVEC(MAXCHN,MAXACC),NOCHAN,NOSUB,
      *        1SUBCAT(MAXSUB),SUBPOP(MAXSUB),CATKNT(MAXCAT),CATTH(MAXCAT),NODO,
      *        2NODU,NOTH,DOTCAT(NDOTS),DOTCLU(NDOTS)
      *   C*
0010  *         INTEGER ADATES,SUNAZ,ANALST,FLDDAY,DOTDAY,PDATE1,TDATE1
0011  *         INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RANDOM,GRID
0012  *         BYTE DELFLG,NOACQ,SOILGR,SUNEL,NSTART,NTYPE1,ALP,ALPO
0013  *         BYTE PCTCT,PCTCTO,VAR,VARO,DLABEL,TYPE
0014  *         COMMON/COM2/ISEG,DELFLG,NOACQ,ADATES(2,MAXACD),SOILGR(MAXACD),
      *        1SUNEL(MAXACD),SUNAZ(MAXACD),IMDATE(2),ANALST(5),FLDDAY(2),
      *        2DOTDAY(2),NSTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      *        3PDATE3(2),TDATE3(2),NOCAT,CATNAM(MAXCAT),ALP(MAXCAT),ALPO,
      *        4          PCTCT(MAXCAT),PCTCTO,VAR(MAXCAT),VARO
      *   C*
0015  *         INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      *        1UFLAG4
0016  *         INTEGER PFLAG,DSKMNT
0017  *         COMMON/COM3/PFLAG,DSKMNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
      *        1,UFLAG2,UFLAG3,UFLAG4,NEFLAG(MAXSUB)
      *   C*
0018  *         INTEGER TX1,TY1,TX2,TY2,ACDISP,G,B,DTWIND,DOTARY,GMIN,GMAX,FUL
0019  *         INTEGER SPWIND,CLAWND,CLUWND
0020  *         COMMON/COM4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACDISP(2),III(4),G(4),
      *        1B(4),DTWIND(5,NODTWD),SPWIND(5,NOSPWD),IMWIND(4),NUMDOT,
      *        2DOTARY(NDOTS),GMIN,GMAX,FUL(2,7),CLAWND(8),CLUWND(8)
0021  *         COMMON/COM5/DISKID,RANDOM(NDOTS),GRID(NDOTS),DLABEL(NDOTS),
      *        1TYPE(NDOTS),RECLOC
0022            DIMENSION REQ4 (5)
0023            DIMENSION REQ8 (10)
0024            DIMENSION REQ (20)
0025            DIMENSION W(37)
0026            DIMENSION GPT(17)
0027            DATA REQ (1)/6RINIT  /
0028            DATA REQ (2)/6RFUL=13/
0029            DATA REQ (3)/6RFLDREF/
0030            DATA REQ (4)/6RDUMMY /
0031            DATA REQ (5)/6RACLLAP/
0032            DATA REQ (6)/6RCLUPIS/
0033            DATA REQ (7)/6RRECPRO/
0034            DATA REQ (8)/6RDUMMY /
```

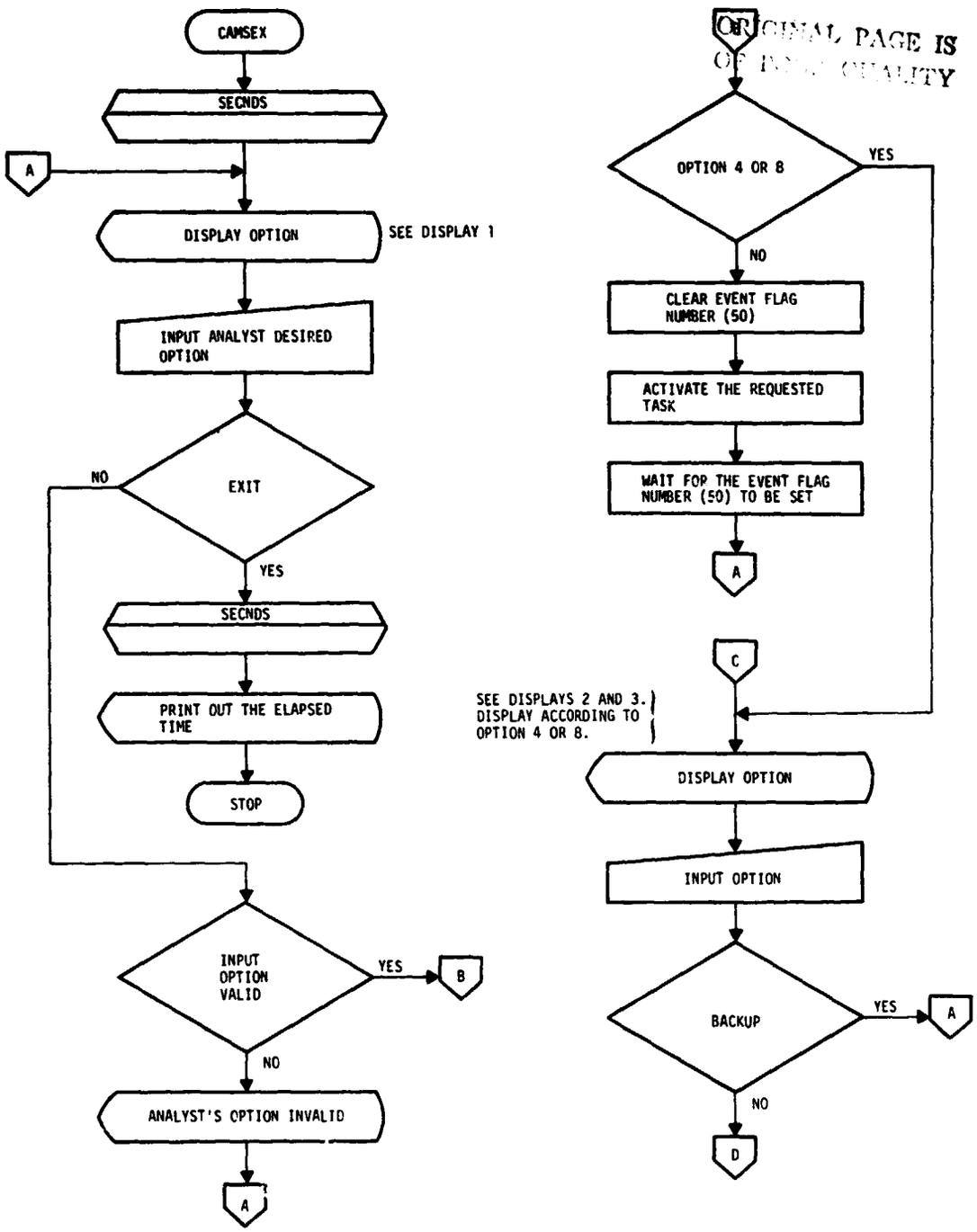
ORIGINAL PAGE IS
OF POOR QUALITY

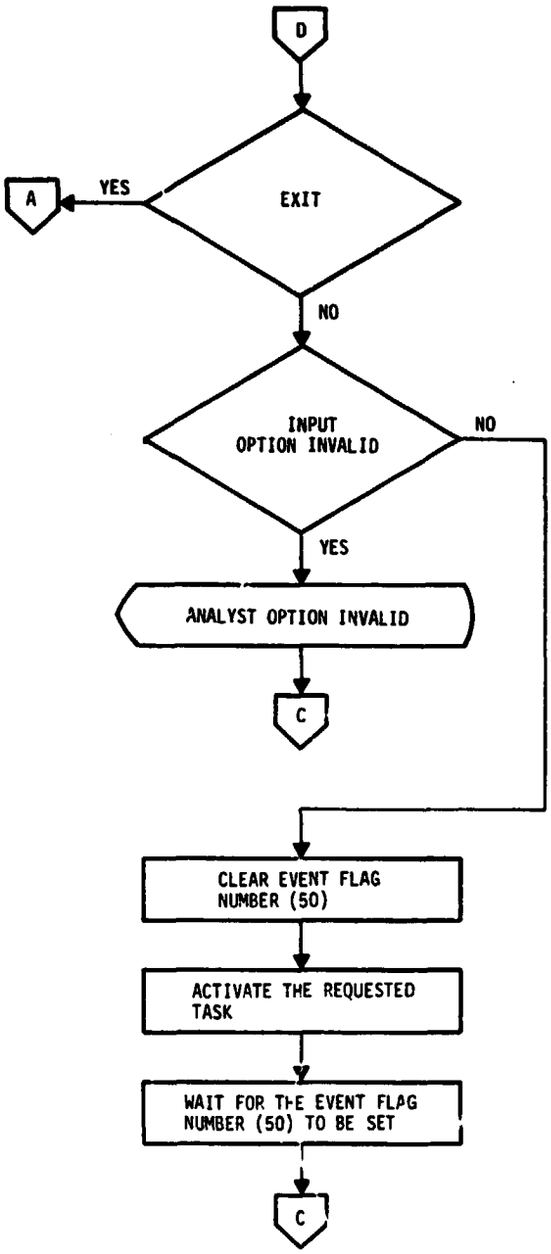
```
FØRTRAN IV-PLUS V02-04          12126145    29-AUG-77          PAGE 2
CAMSEX,FTN____ /TRIBLØCKS/WB
0035            DATA REQ (9)/6HPPMUPD/
0036            DATA REQ (10)/6RCAPTST/
0037            DATA REQ4(1)/6RD2TPVR/
0038            DATA REQ4(2)/6RSCPL2T/
0039            DATA REQ4(3)/6RD2TPFR/
0040            DATA REQ8(1)/6RD2TRPT/
0041            DATA REQ8(2)/6RBIASCR/
0042            DATA REQ8(3)/6RCLUPT/
0043            DATA REQ8(4)/6RFLDRPT/
0044            II=1
0045            CALL ELAPSE(II)
        C
        C       INITIALIZE I-100 GPT HARDWARE
        C
        ?
0046              CALL ATTACH
0047            DØ 80 J=1,17
0048         80 GPT(J)=0
0049            DØ 90 J=1,17,5
0050         90 GPT(J)=255
0051            CALL INXFM(GPT)
0052            CALL WAIT
0053              CALL DETACH
0054      1     CØNTINUE
0055            CALL ØUTPUT (27,12)
0056            WRITE(6,3000)
0057      3000  FØRMAT('1',5X, '***CAMS I-100 CØNTRØL PRØGRAM***' ,/)
0058            WRITE(6,1001)
0059      1001  FØRMAT('  1 INITIATE SEGMENT ANALYSIS')
0060            WRITE(6,1002)
0061      1002  FØRMAT('  2 IMAGE DISPLAY')
0062            WRITE(6,1003)
0063      1003  FØRMAT('  3 FIELD DEFINITIØN')
0064            WRITE(6,1004)
0065      1004  FØRMAT('  4 DET PRØCESSING')
0066            WRITE(6,1005)
0067      1005  FØRMAT('    *DAT GRØUP CRØSSHAIR ØVERLAYS')
0068            WRITE(6,1006)
0069      1006  FØRMAT('    *DAT GRØUP SCATTER PLØTS')
0070            WRITE(6,1007)
0071      1007  FØRMAT('    *DAT LABELLING')
0072            WRITE(6,1008)
0073      1008  FØRMAT('  5 AUTØMATIC CLUSTER LABELLING')
0074            WRITE(6,1009)
0075      1009  FØRMAT('  6 CLUSTER DISPLAY/RELABEL')
0076            WRITE(6,1010)
0077      1010  FØRMAT('  7 CLASSIFICATIØN CØRRECTIØN/MAP DISPLAY')
0078            WRITE(6,1011)
0079      1011  FØRMAT('  8 REPØRTS')
0080            WRITE(6,1012)
0081      1012  FØRMAT('    *DAT DATA REPØRT')
0082            WRITE(6,1013)
0083      1013  FØRMAT('    *BIAS CØRRECTIØN/CLASSIFICATIØN SUMMARY')
0084            WRITE(6,1014)
0085      1014  FØRMAT('    *CLUSTER REPØRTS')
0086            WRITE(6,1015)
```

```
0087     1015  FØRMAT('     *FIELD DATA REPØRT')
0088           WRITE(6,1017)
0089     1017  FØRMAT('  9 DATA BASE UPDATE')
0090           WRITE(6,1032)
0091     1032  FØRMAT('     *PERMANENT DATA BASE UPDATE')
0092           WRITE(6,1033)
0093     1033  FØRMAT('     *CARD IMAGE FILE FØR ØFFLØAD DØT DATA, DØ/DU FIELD')
0094           WRITE(6,1020)
0095     1020  FØRMAT(' 99 CAMS TEST')
0096           WRITE(6,1021)
0097     1021  FØRMAT('  X EXIT')
0098     100   WRITE(6,2000)
0099     2000  FØRMAT(/, '$?' )
0100           CALL ØUTPUT(7)
0101           READ(6,2001)W
0102     2001  FØRMAT(37A2)
0103           CALL FRØNT(W,37)
0104           IF(W(1).EQ.'X ') GØ TØ 150
0105           K=0
0106           CALL INTFF(K,W,37,N)
0107           IF(K.GE.37)GØ TØ 100
0108           IF(N.LE.0)GØ TØ 100
0109           GØ TØ 200
0110     150   CØNTINUE
0111           II=2
0112           CALL ELAPSE(II)
0113           STØP
0114     200   IF(N.NE.99)GØ TØ 250
0115           N=10
0116     250   IF(N.GT.10) GØ TØ 100
0117           IF(N.EQ.4) GØ TØ 300
0118           IF(N.EQ.8) GØ TØ 400
0119           CALL CLREF(50)
0120           CALL REQUES(REQ(N))
0121           CALL WAITFR(50)
0122           GØ TØ 1
0123     300   CØNTINUE
0124           CALL ØUTPUT (27,12)
0125           WRITE(6,1040)
0126     1040  FØRMAT('$',5X, '***CAMS I-100 DØT PRØCESSING***' ,/)
0127           WRITE(6,1022)
0128     1022  FØRMAT('    1 DØT GRØUP CRØSSHAIR ØVERLAYS')
0129           WRITE(6,1023)
0130     1023  FØRMAT('    2 DØT GRØUP SCATTER PLØTS')
0131           WRITE(6,1024)
0132     1024  FØRMAT('    3 DØT LABELLING')
0133     301   WRITE(6,3001)
0134     3001  FØRMAT(/, '$?' )
0135           CALL ØUTPUT(7)
0136           READ(6,3002)W
0137     3002  FØRMAT(37A2)
0138           CALL FRØNT(W,37)
0139           IF(W(1).EQ.'B ') GØ TØ 1
0140           IF(W(1).EQ.'X ') GØ TØ 1
0141           K=0
0142           CALL INTFF(K,W,37,N)
```

```
CAMSEX,FTN      /TR:BLOCKS/WR
0143            IF(N.LE.0) GO TO 302
0144            IF(N.GT.3) GO TO 302
0145            CALL CLREF(50)
0146            CALL REQUES(REQ4(N))
0147            CALL WAITFR(50)
0148            GO TO 300
0149    400     CONTINUE
0150            CALL OUTPUT (27,12)
0151            WRITE(6,1041)
0152    1041    FORMAT('$',5X, '***CAMS I-100 REPORTS***' ,/)
0153            WRITE(6,1025)
0154    1025    FORMAT('    1 DBT DATA REPORT')
0155            WRITE(6,1026)
0156    1026    FORMAT('    2 BIAS CORRECTION/CLASSIFICATION SUMMARY')
0157            WRITE(6,1027)
0158    1027    FORMAT('    3 CLUSTER REPORTS')
0159            WRITE(6,1028)
0160    1028    FORMAT('    4 FIELD DATA REPORT')
0161    401     WRITE(6,4000)
0162    4000    FORMAT(/, '$?' )
0163            CALL OUTPUT(7)
0164            READ(6,4001)W
0165    4001    FORMAT(37A2)
0166            CALL PRINT(W,37)
0167            IF(W(1).EQ.'P ') GO TO 1
0168            IF(W(1).EQ.'X ') GO TO 1
0169            K=1
0170            CALL INTER(N,W,37, )
0171            IF(N.LE.0) GO TO 402
0172            IF(N.GT.5) GO TO 402
0173            CALL CLREF(50)
0174            CALL REQUES(REQ5(N))
0175            CALL WAITFR(50)
0176            GO TO 400
0177    302     WRITE(6,1030)
0178    1030    FORMAT(' INVALID OPTION')
0179            GO TO 301
0180    402     WRITE(6,1031)
0181    1031    FORMAT(' INVALID OPTION')
0182            GO TO 401
0183            END
```

ORIGINAL PAGE IS
OF POOR QUALITY




D
EXIT
YES
A
NO
INPUT
OPTION INVALID
NO
YES
ANALYST OPTION INVALID
C
CLEAR EVENT FLAG
NUMBER (50)
ACTIVATE THE REQUESTED
TASK
WAIT FOR THE EVENT FLAG
NUMBER (50) TO BE SET
C

```
MFØRTRAN IV-PLUS V02-04                15:32:24    14-JUL-77          PAGE 1
INIT.FTN        /TR:BLPCKS/WR
0001            IMPLICIT INTEGER(A-Z)
0002            INCLUDE 'SYS[300,3]CAMSCØMØN.INC'
0003  *         INCLUDE 'SYS[300,3]CAMSPARAM.INC'
0004  *         PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      *        1,MAXV=11,NDØTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *        2NØSPWD=6,NØDTWD=10
0005  *         EQUIVALENCE (C1,ACDATE),(C2,ISEG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0006  *         INTEGER C1(469),C2(256),C3(71),C4(348),C5(629)
      *   C*
0007  *         INTEGER ACDATE,SUBCAT,SUBPØP,CATKNT,CATTH
0008  *         BYTE CHNVEC,NØCHAN,NØSUB,DØTCAT,DØTCLU
0009  *         COMMON/CØM1/ACDATE(2,MAXACC),CHNVEC(MAXCHN,MAXACC),NØCHAN,NØSUB,
      *        1SUBCAT(MAXSUB),SUBPØP(MAXSUB),CATKNT(MAXCAT),CATTH(MAXCAT),NØDØ,
      *        2NØDU,NØTH,DØTCAT(NDØTS),DØTCLU(NDØTS)
      *   C*
0010  *         INTEGER ADATES,SUNAZ,ANALST,FLDDAY,DØTDAY,PDATE1,TDATE1
0011  *         INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RANDØM,GRID
0012  *         BYTE DELFLG,NØACQ,SØILGR,SUNEL,NSTART,NTYPE1,ALP,ALPØ
0013  *         BYTE PCTCT,PCTCTØ,VAR,VARØ,DLABEL,TYPE
0014  *         COMMON/CØM2/ISEG,DELFLG,NØACQ,ADATES(2,MAXACD),SØILGR(MAXACD),
      *        1SUNEL(MAXACD),SUNAZ(MAXACD),IMDATE(2),ANALST(5),FLDDAY(2),
      *        2DØTDAY(2),NSTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      *        3PDATE3(2),TDATE3(2),NØCAT,CATNAM(MAXCAT),ALP(MAXCAT),ALPØ,
      *        4          PCTCT(MAXCAT),PCTCTØ,VAR(MAXCAT),VARØ
      *   C*
0015  *         INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      *        1UFLAG4
0016  *         INTEGER PFLAG,DSKMNT
0017  *         COMMON/CØM3/PFLAG,DSKMNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
      *        1,UFLAG2,UFLAG3,UFLAG4,NEWLAB(MAXSUB)
      *   C*
0018  *         INTEGER TX1,TY1,TX2,TY2,ACDISP,G,B,DTWIND,DØTARY,GMIN,GMAX,FUL
0019  *         INTEGER SPWIND,CLAWND,CLUWND
0020  *         COMMON/CØM4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACDISP(2),III(4),G(4),
      *        1B(4),DTWIND(5,NØDTWD),SPWIND(5,NØSPWD),IMWIND(4),NUMDØT,
      *        2DØTARY(NDØTS),GMIN,GMAX,FUL(2,7),CLAWND(8),CLUWND(8)
0021  *         COMMON/CØM5/DISKID,RANDØM(NDØTS),GRID(NDØTS),DLABEL(NDØTS),
      *        1TYPE(NDØTS),RECLØC
0022            BYTE DTSDA1(27),DTSDA0(22),FLDATA(27),FLDTMP(24)
0023            BYTE DFX,DLF,DFB
0024            BYTE XTCLAM(27),XPCLAM(27),XTCLUM(27),TITLE(45),
                1XPCLUM(27),XTSTDT(27),XPSTDT(27),STATMP(25),ØNETWØ,KEYIN(20)
0025            BYTE SEGN(4),PUNCHR(10),TEMP,ADIGIT(10),DTGXYT(20),SCTXYT(20)
0026              BYTE TMPCLM(27),TMPCLU(27),VERTX(11),VERTY(11),FLDNAM(6)
0027              LØGICAL*1 KPAT01(52)
0028            REAL TSTART,TELPSD,KPAT02(36)
0029            EQUIVALENCE(IPNCHR,PUNCHR),(ØNETWØ,KEYIN(1))
0030              EQUIVALENCE(DØTDA(1),KPAT01(1))
0031              EQUIVALENCE(KPAT02(1),DATA(1))
0032            DIMENSIØN DØTDA(26),MAXDV(4),MINDV(4),IPNCHR(5),FLDBUF(15)
0033            DIMENSIØN DATA(72),DPI(3),XDØTGS(NDØTS)
0034            DIMENSIØN DAFLAG(6),RED(5),GRD(5),CLAMED(5),CLUMED(5),STATED(5),
                1DØDULU(5),DØTDLU(5),CLUSTT(26)
0035            DATA TMPCLM/'D','B','Ø','!','[','3','Ø','Ø',',',',','1',']','C','L',
                1'A','S','S','M','A','P',',',',','T','M','P',']','1',0,0/
0036            DATA TMPCLU/'D','B','Ø','!','[','3','Ø','Ø',',',',','1',']','C','L',
```

```
INIT.FTN        /TRIBLØCKS/WR
               1'U','S','T','E','R','M','P','.','.','T','M','P','.','1',0/
0037            DATA XTCLAM/'D','B','2','.','[','3','0','0','.','.','3','0','0',']',
               1'X','X','X','X','T','C','L','A','S','.','.','M','A','P',0/
0038            DATA XPCLAM/'D','B','2','.','[','3','0','0','.','.','3','0','0',']',
               1'X','X','X','X','P','C','L','A','S','.','.','M','A','P',0/
0039            DATA XTCLUM/'D','B','2','.','[','3','0','0','.','.','3','0','0',']',
               1'X','X','X','X','T','C','L','U','S','.','.','M','A','P',0/
0040            DATA XPCLUM/'D','B','2','.','[','3','0','0','.','.','3','0','0',']',
               1'X','X','X','X','P','C','L','U','S','.','.','M','A','P',0/
0041            DATA XTSTDT/'D','B','2','.','[','3','0','0','.','.','3','0','0',']',
               1'X','X','X','X','T','S','T','A','T','.','.','D','A','T',0/
0042            DATA XPSTDT/'D','B','2','.','[','3','0','0','.','.','3','0','0',']',
               1'X','X','X','X','P','S','T','A','T','.','.','D','A','T',0/
0043            DATA FLDATA/'D','B','2','.','[','3','0','0','.','.','3','0','0',']','X
               1','X','X','X','F','I','E','L','D','.','.','D','A','T',0/
0044            DATA DTSDA1/'D','B','2','.','[','3','0','0','.','.','3','0','0',']','X
               1','X','X','X','D','D','0','T','S','.','.','D','A','T',0/
0045            DATA DTSDA0/'D','B','0','.','[','3','0','0','.','.','1','.',']','D',
               1  '0','T','S','.','.','T','M','P','.','.','1',0/
0046            DATA FLDTMP/'D','B','0','.','[','3','0','0','.','.','1','.',']','F',
               1  'I','E','L','D','S','.','.','T','M','P','.','.','1',0/
0047            DATA DTGXYT/'[','3','0','0','.','.','1','.',']','D','0','T','G','X','Y',
               1',','T','M','P','.','.','1',0/
0048            DATA SCTXYT/'[','3','0','0','.','.','1','.',']','S','C','A','T','X','Y',
               1',','T','M','P','.','.','1',0/
0049            DATA STATMP/'D','B','0','.','[','3','0','0','.','.','1','.',']','S',
               1'T','A','T','F','I','L','E','.','T','M','P','.','.','1',0/
0050            DATA CLUSTT/'D','B','0','.','[','3','0','0','.','.','1','.',']',
               1'C','L','U','S','T','A','T','S','.','.','T','M','P','.','.','1',0/
0051            DATA TITLE/'I','N','.','T','.','A','T','E',' ','S','E','G','M',
               1'E','N','T',' ','A','N','A','L','Y','S','I','S',' ','V','E','R',
               2'S','.','0','N',' ','1','.','.',' ','M','A','Y',' ','1','9','7','7'/
0052            DATA ADIGIT/'0','1','2','3','4','5','6','7','8','9'/
0053            DATA DFX,DFB,IN,I0,K0,IT/'Y',' ',5,6,4,4/
0054            DATA XD0TGS/ND0TS*0/
0055            II=1
0056            CALL ELAPSE(II)
0057            PAGE=1
0058            CALL CSGDPH(5,PAGE,TITLE,45)
0059            MD0TS=ND0TS
0060            MLIN=NLIN
0061            MPIX=NPIX
0062            FT=7
0063            FD=9
0064            DFU=8
0065            LRW=204
0066            NR=100
0067            0PEN(UNIT=DFU,TYPE='UNKN0WN',NAME=DTGXYT,REC0RDSIZE=ND0TS,MAXREC=
               11,ACCESS='SEQUENTIAL',F0RM='UNF0RMATTED')
0068            WRITE(DFU)XD0TGS,XD0TGS
0069            CL0SE(UNIT=DFU,DISP0SE='SAVE')
0070            20PEN(UNIT=DFU,TYPE='UNKN0WN',NAME=SCTXYT,REC0RDSIZE=ND0TS,MAXREC=
               1N0SPWD,ACCESS='DIRECT')
0071            D0 4 I=1,N0SPWD
0072            WRITE(DFU'I)XD0TGS,XD0TGS
0073    4       C0NTINUE
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
FØRTRAN IV-PLUS V02-04          15:32:24    14-JUL-77          PAGE 3
INIT.FTN        /TR:BLØCKS/WR
0074            CLØSE(UNIT=DFU,DISPØSE='SAVE')
0075            ØPEN(UNIT=DFU,TYPE='ØLD',NAME='DB0:[300,300]DIRFILE.DAT',
               1RECØRDSIZE=128,MAXREC=200,ACCESS='DIRECT')
        C
        CC
        C                                          A
        C
        C
0076    10      TYPE 901
0077            ACCEPT 800,KEYIN
0078            CALL FRØNT(KEYIN,20)
0079            IF(ØNETWØ.EQ.'R')GØ TØ 20
0080            IF(ØNETWØ.EQ.'N')GØ TØ 1
0081            IF(ØNETWØ.EQ.'X')GØ TØ 90
0082            IF(ØNETWØ.EQ.'B')GØ TØ 90
0083            TYPE 801
0084            GØ TØ 10
        C
        CC
        C                                          B
        C
        C
0085    1       TYPE 902
0086            ACCEPT 800,(KEYIN(I),I=1,4)
0087            CALL FRØNT(KEYIN,20)
0088            IF(ØNETWØ.EQ.'X')GØ TØ 90
0089            IF(ØNETWØ.EQ.'B')GØ TØ 10
0090            CALL INTLZE
0091            DØ 3 I=1,4
0092            SEGN(I)=KEYIN(I)
0093            TEMP=SEGN(I)
0094            DØ 2 K=1,10
0095            IF(TEMP    .NE.ADIGIT(K))GØ TØ 2
0096            FLDATA(I+13)=TEMP
0097            DTSDA1(I+13)=TEMP
0098            XTCLAM(I+13)=TEMP
0099            XPSTRT(I+13)=TEMP
0100            XPCLAM(I+13)=TEMP
0101            XTCLUM(I+13)=TEMP
0102            XPCLUM(I+13)=TEMP
0103            XTSTRT(I+13)=TEMP
0104            GØ TØ 3
0105    2       CØNTINUE
0106            TYPE 802,SEGN
0107            GØ TØ 1
0108    3       CØNTINUE
0109            DECØDE(4,821,SEGN)ISEG
        C
0110    31      CALL DSKCHK(ISEG,PTR,DSKPT,FLAG)
0111            DISKID=DSKPT
0112            IF(FLAG.LE.0)GØ TØ 40
0113            IF(FLAG.EQ.1)GØ TØ 5
0114            TYPE 905,SEGN
0115            GØ TØ 10
0116    5       TYPE 903,SEGN,SEGN
0117            ACCEPT 800,KEYIN
```

```
0118            CALL FR0NT(KEYIN,20)
0119            IF(0NETW0.EQ.'Y')G0 T0 52
0120            IF(0NETW0.NE.'N')G0 T0 53
0121            IF(0NETW0.EQ.'X')G0 T0 90
0122            IF(0NETW0.EQ.'B')G0 T0 1
0123            G0 T0 10
0124     52     TYPE 815
0125            ACCEPT 800,KEYIN
0126            CALL FR0NT(KEYIN,20)
0127            IF(0NETW0.EQ.'X')G0 T0 90
0128            IF(0NETW0.EQ.'B')G0 T0 5
0129            G0 T0 31
0130     53     TYPE 816
0131            G0 T0 5
0132     20     C0NTINUE
0133            0PEN(UNIT=FD,NAME='SY:[300,1]GL0BAL.TMP11',TYPE='UNKN0WN',
               1ERR=78,F0RM='UNF0RMATTED')
0134            ISW=1
0135            READ(FD)C1
0136            READ(FD)C2
0137            READ(FD)C3
0138            READ(FD)C4
0139            READ(FD)C5
0140            CL0SE(UNIT=FD)
0141            G0 T0 41
0142     40     C0NTINUE
0143            RECL0C=PTR
0144            FIND(DFU'PTR)
0145            READ(DFU'PTR,ERR=77)C2
0146            ISW=2
0147     41     CALL CSGDPH(5,PAGE,TITLE,45)
0148            TYPE 932,ISEG,DSKPT
0149            TYPE 933,ANALST
0150            TYPE 934
0151            IDLF=DELFLG
0152            D0 42 KKK=1,N0ACQ
0153            DLF=DFR
0154            IF(IDLF.EQ.KKK)DLF='*'
0155            TYPE 935,((ADATES(JJJ,KKK),JJJ=1,2),DLF)
0156     42     C0NTINUE
0157            TYPE 936,PDATE1,IMDATE
0158            TYPE 937,TDATE1,TDATE2,TDATE3
0159            TYPE 938,FLDDAY
0160            TYPE 939,D0TDAY
0161            IF(ISW.EQ.1)G0 T0 93
0162     46     C0NTINUE
0163            TYPE 914
0164            ACCEPT 800,KEYIN
0165            CALL FR0NT(KEYIN,20)
0166     43     C0NTINUE
               C
               C
               C                                        FLOW CHART PAGE 2
               C
               C
0167            IF(0NETW0.EQ.'X')G0 T0 90
```

```
INIT,FTN        /TR:BLØCKS/WR
0168            IF(ØNETWØ.EQ.'R')GØ TØ 10
0169            IF(ØNETWØ.EQ.'C')GØ TØ 60
0170            IF(ØNETWØ.EQ.'H')GØ TØ 60
0171            GØ TØ 46
0172    60      CØNTINUE
0173            CALL CSØDPH(5,PAGE,TITLE,45)
0174            TYPE 906
0175            ACCEPT 800,PUNCHR
0176            CALL FRØNT(PUNCHR,10)
0177            DØ 69 I=1,5
0178    69      ANALST(I)=PNCHR(I)
0179    61      TYPE 907
0180            ACCEPT 800,KEYIN
0181            CALL FRØNT(KEYIN,20)
0182            IF(ØNETWØ.EQ.'P')GØ TØ 65
0183            IF(ØNETWØ.EQ.'T')GØ TØ 62
0184            IF(ØNETWØ.EQ.'X')GØ TØ 90
0185            IF(ØNETWØ.EQ.'B')GØ TØ 60
0186            TYPE 914
0187            GØ TØ 61
0188    62      CØNTINUE
0189            DØ 63 I=1,2
0190            IF(TDATE1(I).NE.TDATE2(I))GØ TØ 64
0191            IF(TDATE1(I).NE.TDATE3(I))GØ TØ 64
0192    63      CØNTINUE
0193            PFLAG=0
0194            GØ TØ 66
0195    64      TYPE 908
0196            GØ TØ 90
0197    65      PFLAG=1
0198    66      CØNTINUE
0199            IF(FLDDAY(1).NE.0)GØ TØ 50
0200            EFLAG3=0
0201            TYPE 909,SEGN
0202            GØ TØ 30
0203    50      ØPEN(UNIT=FD,NAME=FLDATA,ACCESS='SEQUENTIAL',TYPE='ØLD',MAXREC=
               1MAXFLD,FØRM='UNFØRMATTED',ERR=79)
0204            ØPEN(UNIT=FT,NAME=FLDTMP,ACCESS='SEQUENTIAL',TYPE='NEW',MAXREC=
               1MAXFLD,FØRM='UNFØRMATTED',ERR=80)
0205            READ(FD,ERR=71)NØFLD
0206            WRITE(FT)NØFLD
0207            DØ 70 I=1,NØFLD
0208            READ(FD,END=70,ERR=71)FLDBUF
0209            WRITE(FT,ERR=71)FLDBUF
0210    70      CØNTINUE
0211            CLØSE(UNIT=FT, DISPØSE='SAVE')
0212            CLØSE(UNIT=FD, DISPØSE='SAVE')
0213    68      EFLAG3=1
0214            TYPE 911
        C                       FLØWCHART PAGE 3
0215    30      CØNTINUE
0216              GMAX=-9999
0217              GMIN=9999
0218            IF(DØTDAY(1).NE.0)GØ TØ 91
0219            TYPE 913,SEGN
0220            GØ TØ 90
```

```
0221     51     OPEN(UNIT=FD,NAME='DTSDA1',TYPE='OLD',ACCESS='DIRECT'
              1,MAXREC=ND0TS,RECORDSIZE=13)
0222            OPEN(UNIT=FT,NAME='DTSDA0',TYPE='NEW',ACCESS='DIRECT',
              1RECORDSIZE=13,MAXREC=ND0TS)
0223            D0 29 I=1,ND0TS
0224            READ(FD'I,ERR=36)D0TDA
0225              IF(KPAT01(3) .EQ. -9) KPAT01(3)=0
0226              IF(KPAT01(4) .EQ. -9) KPAT01(4)=0
0227            WRITE(FT'I)D0TDA
0228            K=3
0229              P=4
0230            TYPE(I)=KPAT01(3)
0231            DLABEL(I)=KPAT01(4)
0232              D0 24 O=1,6
0233            D0 23 J=1,4
0234            K=K+1
0235              IF(KPAT01(K+1) .EQ. -9) G0 T0 29
0236            IDV=IBYTE(K,KPAT01)
0237              GMAX=MAX(GMAX,IDV)
0238              GMIN=MIN(GMIN,IDV)
0239        23    CONTINUE
0240              P=P+1
0241              IDV=D0TDA(P)
0242            IF(IDV.EQ.-2057)G0 T0 24
0243              GMAX=MAX(GMAX,IDV)
0244              GMIN=MIN(GMIN,IDV)
0245              P=P+1
0246              IDV=D0TDA(P)
0247              GMAX=MAX(GMAX,IDV)
0248              GMIN=MIN(GMIN,IDV)
0249              P=P+2
0250              K=K+3
0251        24    CONTINUE
0252     29     CONTINUE
0253            CLOSE(UNIT=FT,DISPOSE='SAVE')
0254            CLOSE(UNIT=FD,DISPOSE='SAVE')
0255            G0 T0 32
0256     36     UFLAG3=0
0257            TYPE 912,SEGN
0258            G0 T0 90
0259     32     CONTINUE
0260            TYPE 913
0261            IF(PFLAG.EQ.0)G0 T0 33
0262            IF(PDATE1(1).NE.0)G0 T0 34
0263            TYPE 804
0264            G0 T0 90
0265     33     CONTINUE
0266            IF(TDATE1(1).NE.0)G0 T0 81
0267            TYPE 820,XTCLAM
0268            G0 T0 82
0269     81     CALL FTRNFR(XTCLAM,TMPCLM,MLIN,MPIX,MLIN,EFLAG2)
0270            IF(EFLAG2.EQ.1)G0 T0 37
0271            TYPE 803
0272            G0 T0 90
0273     34     CALL FTRNFR(XPCLAM,TMPCLM,MLIN,MPIX,MLIN,EFLAG2)
0274            IF(EFLAG2.EQ.1)G0 T0 37
```

```
INIT,FTN        /TRIAL0CKS/WR
0275            TYPE 805
0276            GO TO 1
0277      37    CONTINUE
          C                     4A
0278            IF(PFLAG.NE.0)GO TO 44
0279            CALL FTRNFR(XTCLUM,TMPCLU,MLIN,MPIX,MLIN,EFLAG1)
0280            IF(EFLAG1.EQ.1)GO TO 54
0281            TYPE 806
0282            GO TO 90
0283      44    CONTINUE
0284            IF(PDATE2(1).NE.-9)GO TO 45
          C       IF(PERMANENT CLUSTER MAP EXISTS)GO TO 45
0285            TYPE 807
0286            GO TO 90
0287      45    CALL FTRNFR(XPCLUM,TMPCLU,MLIN,MPIX,MLIN,EFLAG1)
0288            IF(EFLAG1.EQ.1)GO TO 54
0289            TYPE 808
0290            GO TO 90
          C                         FLOW CHART PAGE 5
0291      54    CONTINUE
0292            OPEN(UNIT=FT,NAME='CLUSTT',TYPE='NEW',ACCESS='DIRECT',
               1MAXREC=MAXCAT,RECORDSIZE=36)
0293            IF(PFLAG.NE.0)GO TO 57
0294            IF(TDATE3(1).LE.0)GO TO 35
0295            OPEN(UNIT=FD,NAME='XTSTDT',TYPE='OLD',ACCESS='SEQUENTIAL',
               1FORM='UNFORMATTED')
0296            READ(FD,ERR=72)C1
          C       Q=1
          C       X=7
          C       Y=17
0297            DO 59 I=1,MAXSUB
0298            Q=1
0299            X=7
0300            Y=17
0301            J=I+1
0302            READ(FD,ERR=73,END=56)DATA
0303            WRITE(FT'I)DATA
0304              DO 101 S=1,4
0305              IF(DATA(Q) .EQ. 0.0)GO TO 59
0306              DO 100 BR=0,X,2
0307              IDV=DAT02(BR)
0308              GMAX=MAX(GMAX,IDV)
0309              GMIN=MIN(GMIN,IDV)
0310        100   CONTINUE
0311              Q=Q+9
0312              X=X+9
0313              Y=Y+18
0314              IDV=DATA(Y)
0315              GMAX=MAX(GMAX,IDV)
0316              GMIN=MIN(GMIN,IDV)
0317              IDV=DATA(Y+1)
0318              GMAX=MAX(GMAX,IDV)
0319              GMIN=MIN(GMIN,IDV)
0320        101   CONTINUE
0321      55    CONTINUE
0322      56    CONTINUE
```

```
                C                    CHECK FLOW CHART
0323                 CLOSE(UNIT=FD,DISPOSE='SAVE')
0324                 CLOSE(UNIT=FT,DISPOSE='SAVE')
0325                 EFLAG5=1
0326                 TYPE 940
0327                 GO TO 90
0328            57   CONTINUE
0329                 IF(PDATE3(1).EQ.-9)GO TO 74
0330                 OPEN(UNIT=FD,NAME=YPSTDT,ACCESS='SEQUENTIAL',TYPE='OLD',
                    1FORM='UNFORMATTED')
0331                 READ(FD,ERR=75)C1
0332                 DO 58 I=1,MAXSUB
0333                 J=I+1
0334                 READ(FD,ERR=76,END=56)DATA
0335                 WRITE(FT'I)DATA
0336            58   CONTINUE
0337                 GO TO 56
0338            71   EFLAG3=0
0339                 TYPE 910
0340                 GO TO 90
0341            72   EFLAG5=0
0342                 TYPE 809
0343                 GO TO 90
0344            73   EFLAG5=0
0345                 TYPE 810
0346                 GO TO  90
0347            74   TYPE 811
0348                 GO TO 82
0349            76   CONTINUE
0350            75   EFLAG5=0
0351                 TYPE 812
0352                 GO TO 90
0353            77   TYPE 817
0354                 GO TO 90
0355            78   TYPE 818
0356                 GO TO 90
0357            79   TYPE 819,FLDATA
0358                 GO TO 90
0359            80   TYPE 819,FLDTMP
0360                 GO TO 90
0361            35   CONTINUE
0362                 TYPE 822
0363            82   CONTINUE
                C                                   PROGRAM EXIT LOGIC
0364            90   CONTINUE
0365                 DO 92 I=1,MAXSUB
0366                 NEWLAB(I)=SUBCAT(I)
0367            92   CONTINUE
0368            93   II=2
0369                 CALL ELAPSE(II)
0370                 TYPE 917
0371                 ACCEPT 800,KEYIN
0372                 CALL PRINT(KEYIN,20)
0373                 IF(KEYIN.EQ.'R')GO TO 1
0374                 IF(KEYIN.EQ.'R')GO TO 91
0375                 IF(KEYIN.EQ.'X')GO TO 91
```

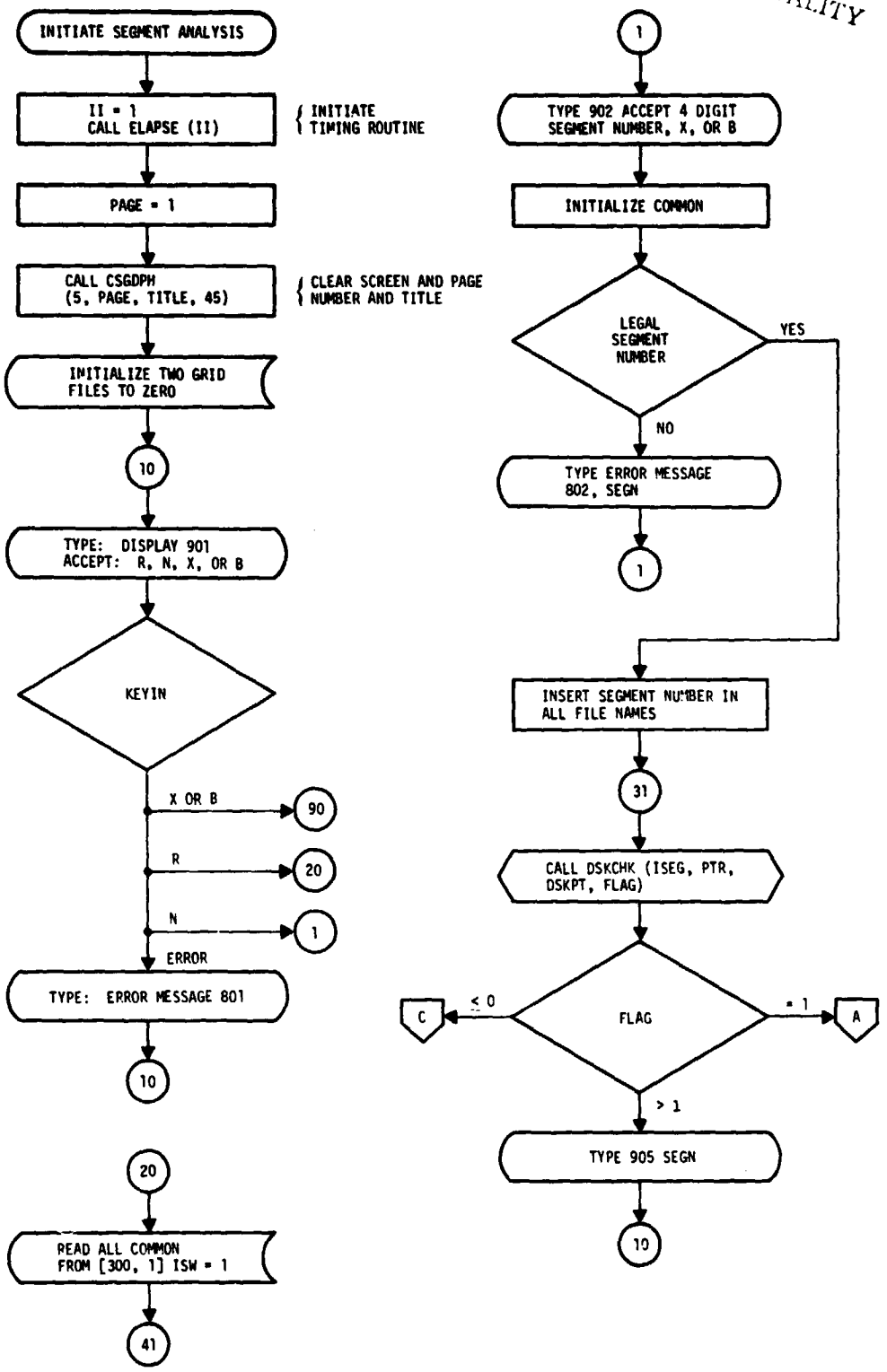
```
0376            GØ TØ 90
0377      91    CØNTINUE
0378            INCLUDE'[300,3]CAMSAVE.INC'
0379  *          ØPEN(UNIT=1,NAME='[300,1]GLØBAL.TMP;1',FØRM='UNFØRMATTED',
      *       1   TYPE='UNKNØWN',ERR=9999)
0380  *          WRITE(1)C1
0381  *          WRITE(1)C2
0382  *          WRITE(1)C3
0383  *          WRITE(1)C4
0384  *          WRITE(1)C5
0385  *          CLØSE(UNIT=1)
0386  *          GØ TØ 9991
0387  *  9999    TYPE 9990
0388  *  9990    FØRMAT(1X,'ØPEN FAILURE ØN [300,1]GLØBAL.TMP--NØ RESTART')
0389  *  9991    CØNTINUE
0390      99    CALL SETEF(50)
0391            STØP
0392     901    FØRMAT(' THE FØLLØWING MØDES FØR INITIATING SEGMENT ANALYSIS ARE '
               1,'AVAILABLE:'/4X,'1. (N)ØRMAL SEGMENT ANALYSIS INITIATIØN'/
               24X,'2. (R)ESTART SEGMENT ANALYSIS AFTER SYSTEM CRASH'/4X,
               3'3. E(X)IT SEGMENT ANALYSIS'/'$INPUT EITHER N, R, ØR X >')
0393     902    FØRMAT('$INPUT SEGMENT NUMBER-FØUR(4) DIGITS REQUIRED >')
0394     903    FØRMAT(' YØU HAVE REQUESTED SEGMENT NØ.',2X,4A1/
               1' THE DISK PACK CØNTAINING DATA FØR THIS SEGMENT IS NØT CURRENTLY
               2MØUNTED'/' DØ YØU WISH THE DISK PACK FØR SEGMENT NØ.',4A1,
               3' TØ BE MØUNTED?'/'$INPUT EITHER  YES  ØR  NØ >')
0395     905    FØRMAT(' YØU HAVE REQUESTED SEGMENT NØ. '4A1,' WHICH IS NØT CURREN
               1TLY IN THE SYSTEM')
0396     906    FØRMAT('$INPUT ANALYST NAME - DØ NØT EXCEED 10 CHARACTERS >')
0397     907    FØRMAT(' DØ YØU DESIRE PERMANENT FILES ØR TEMPØRARY FILES FØR ANAL
               1YSIS '/'$INPUT P ØR T >')
0398     908    FØRMAT(' THE DATES ØF THE TEMPØRARY CLUSTERING MAP AND THE TEMPØRA
               1RY'/' CLASSIFICATIØN MAP AND THE TEMPØRARY STATISTICS FILE DØ NØT
               2MATCH'/' FURTHER PRØCESSING ØF THIS SEGMENT IS PRØHIBITED.')
0399     909    FØRMAT(' FIELDS DEFINITIØN FILE DØES NØT EXIST FØR SEGMENT NØ. '
               14A1)
0400     910    FØRMAT(' AN ERRØR WAS ENCØUNTERED WHILE ATTEMPTING TØ LØAD THE FIELD
               1 DEFINITIØN'/' FILE FØR SEGMENT '4A1,'. FURTHER PRØCESSING ØF THIS
               2 SEGMENT IS' /' PRØHIBITED')
0401     911    FØRMAT(' FIELD DEFINITIØN FILE LØADED SUCCESSFULLY')
0402     912    FØRMAT(' AN ERRØR WAS ENCØUNTERED WHILE ATTEMPTING TØ LØAD ØR '/
               1' BUILD THE DØT DATA FILE FØR SEGMENT '4A1,/' FURTHER PRØCESSING ØF
               2 THIS SEGMENT IS PRØHIBITED')
0403     913    FØRMAT(' DØT DATA FILE LØADED SUCCESSFULLY')
0404     914    FØRMAT('$ØPTIØNS REMAINING ARE (C)ØNTINUE, E(X)IT, (R)ESTART>')
0405     917    FØRMAT('$ (R)ESTART ØR E(X)IT >')
0406     932    FØRMAT(1X,'SEGMENT NØ. = ',I4,4X,'DISK PACK IDENTIFIER = ',I4)
0407     933    FØRMAT('ØSEGMENT LAST WØRKED BY ANALYST ',5A2)
0408     934    FØRMAT('ØTHE FØLLØWING DATA BASE FILES EXIST FØR THIS SEGMENT',
               1//,6X,'IMAGE DATA',6X,'ACQUISITIØN DATE(S)',6X,'DELETE FLAG')
0409     935    FØRMAT(28X,I2,1X,I3,18X,A1)
0410     936    FØRMAT(/5X,'PERMANENT CLASSIFICATIØN RESULTS'/,12X,'REFLECT ERIPS
               1 DATE',6X,I2,X,I3,/12X,'LAST I-100 REWØRK DATE',2X,I2,X,I3)
0411     937    FØRMAT(/5X,'TEMPØRARY CLASSIFICATIØN RESULTS'/
               112X,'CLASSIFICATIØN MAP  ERIPS DATE ',I2,X,I3/
               212X,'CLUSTER MAP',9X,'ERIPS DATE ',I2,X,I3/
```

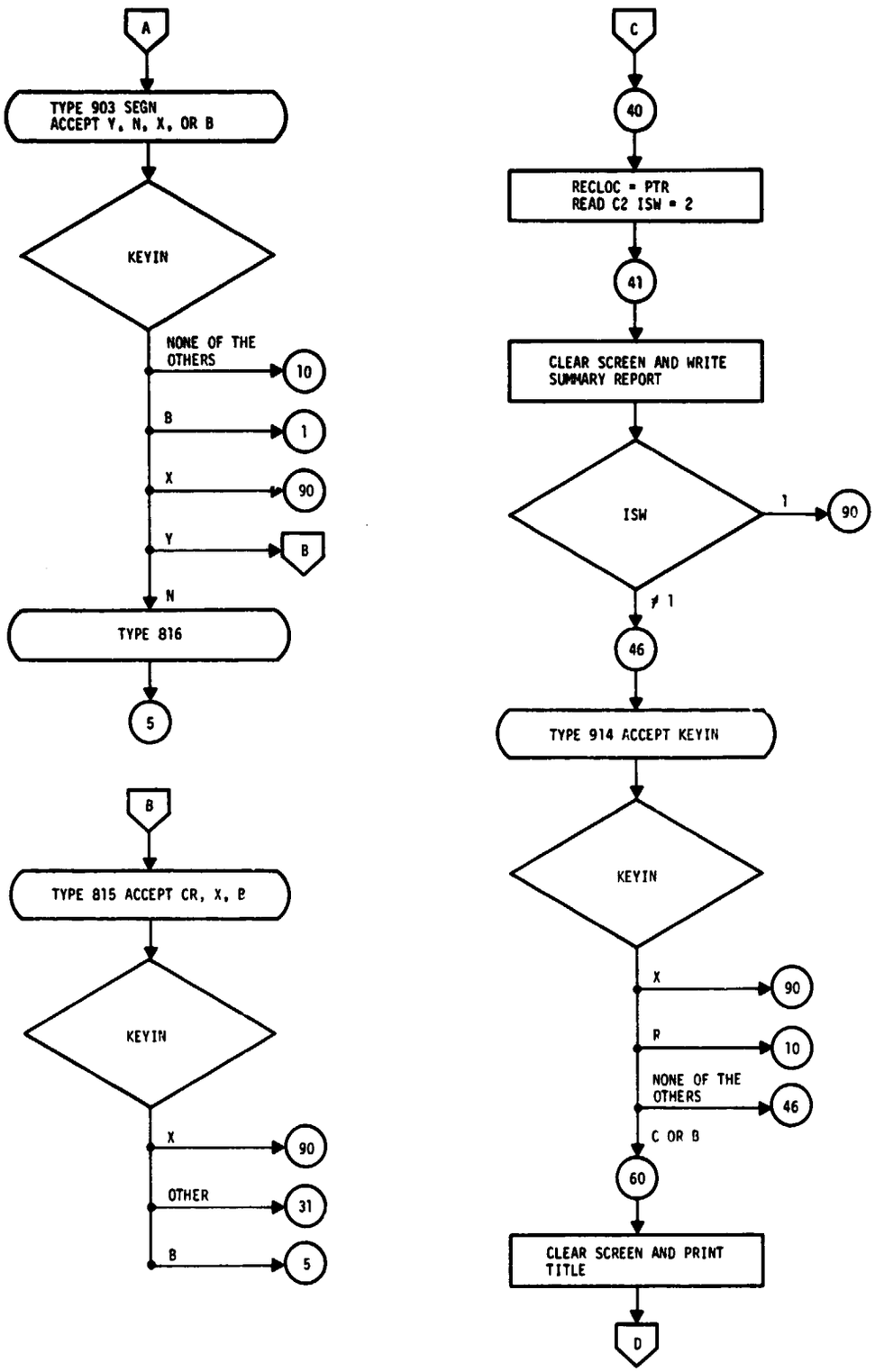

ORIGINAL PAGE IS
OF POOR QUALITY

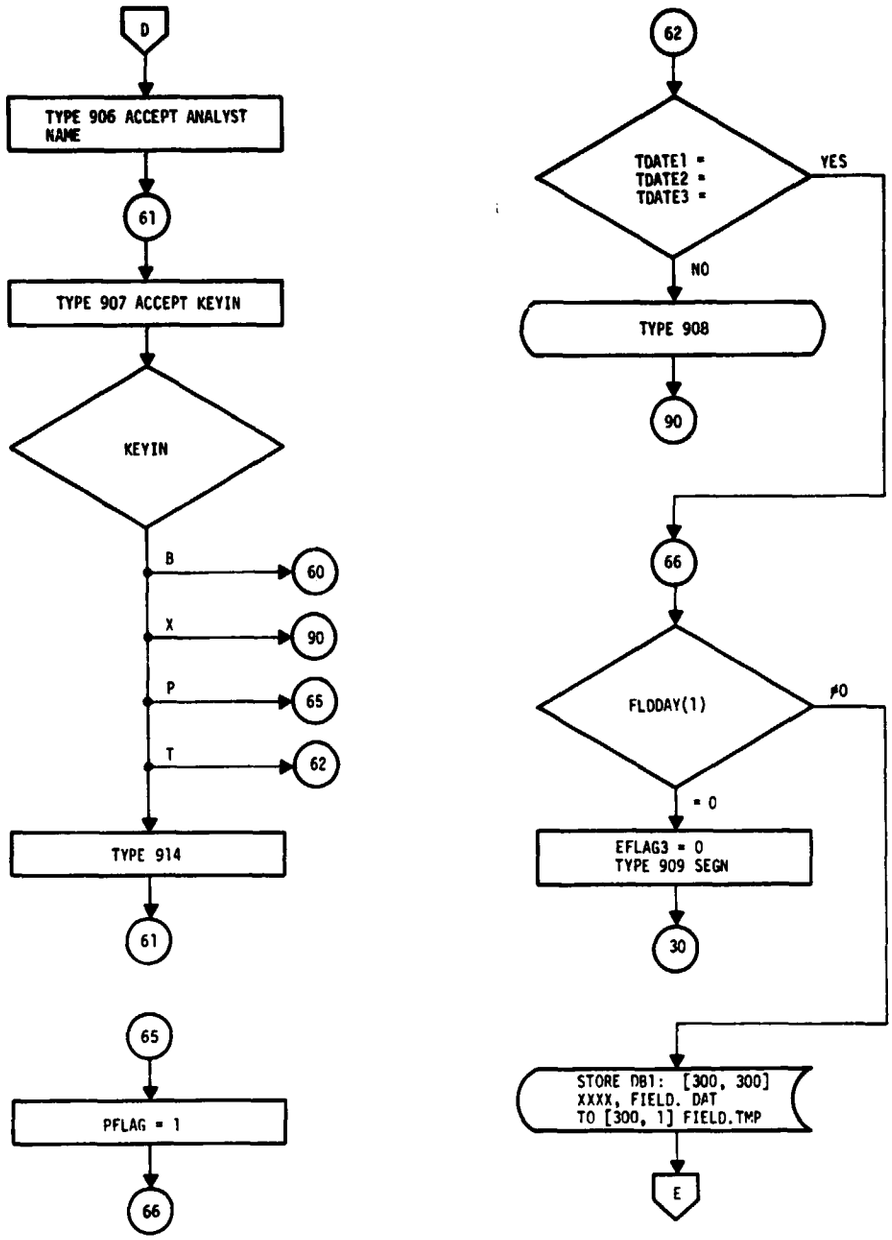
```
            312X,'STATISTICS',10X,'ERIPS DATE ',I2,X,I3)
0412    938   FØRMAT(/' D0/DU FIELDS = LAST UPDATE ',I2,X,I3)
0413    939   FØRMAT(/' DØT DATA',5X,'= LAST UPDATE ',I2,X,I3)
0414    940   FØRMAT('0ALL FILES TRANSFERRED')
0415    800   FØRMAT(80A1)
0416    801   FØRMAT('    INITIATE SEGMENT ANALYSIS MØDE REQUESTED IS ØTHER THAN
            1 N, F, ØR X! TRY AGAIN')
0417    902   FØRMAT(' ',4A1,' ILLEGAL SEGMENT NUMBER -- MUST BE FØUR INTEGERS -
            1 TRY AGAIN')
0418    803   FØRMAT('    ERRØR ENCØUNTERED IN TRANSFER ØF TEMPØRARY CLASSIFICAT
            1IØN MAP')
0419    804   FØRMAT(' PERMANENT CLASSIFICATIØN MAP DØES NØT EXIST')
0420    805   FØRMAT(' ERRØR ENCØUNTERED IN TRANSFER ØF PERMANENT CLASSIFICATIØN
            1 MAP')
0421    806   FØRMAT(' ERRØR ENCØUNTERED IN TRANSFER ØF TEMPØRARY CLUSTER MAP')
0422    807   FØRMAT(' PERMANENT CLUSTER MAP DØES NØT EXIST')
0423    808   FØRMAT(' ERRØR ENCØUNTERED IN TRANSFER ØF PERMANENT CLUSTER MAP')
0424    809   FØRMAT(' ERRØR ENCØUNTERED IN TRANSFER ØF TEMPØRARY STATISTICS FIL
            1E (RECØRD 1)')
0425    810   FØRMAT(' ERRØR ENCØUNTERED IN TRANSFER ØF TEMPØRARY STATISTICS FIL
            1E')
0426    811   FØRMAT(' PERMANENT STATISTICS FILE DØES NØT EXIST')
0427    812   FØRMAT(' ERRØR ENCØUNTERED IN TRANSFER ØF PERMANENT STATICS FILE (
            1RECØRD 1)')
0428    813   FØRMAT(' ERRØR ENCØUNTERED IN TRANSFER ØF PERMANENT STATISTICS FIL
            1E')
0429    814   FØRMAT(' ILLEGAL FILE TYPE--TRY AGAIN')
0430    815   FØRMAT(' MØUNT THE RIGHT DISK AND HIT   CR')
0431    816   FØRMAT(' MUST INPUT (Y)ES ØR (N)Ø')
0432    817   FØRMAT(' ERRØR TRANSFERRING THE DIRECTØRY FILE')
0433    819   FØRMAT('0ØPEN ERRØR ',30A1)
0434    818   FØRMAT('0GLØBAL.TMP ERRØR')
0435    820   FØRMAT(' THIS FILE DØES NØT EXIST > ',30A1)
0436    821   FØRMAT(I4)
0437    822   FØRMAT(' TEMPØRARY STATISTICS FILE DØES NØT EXIST')
0438          END
```

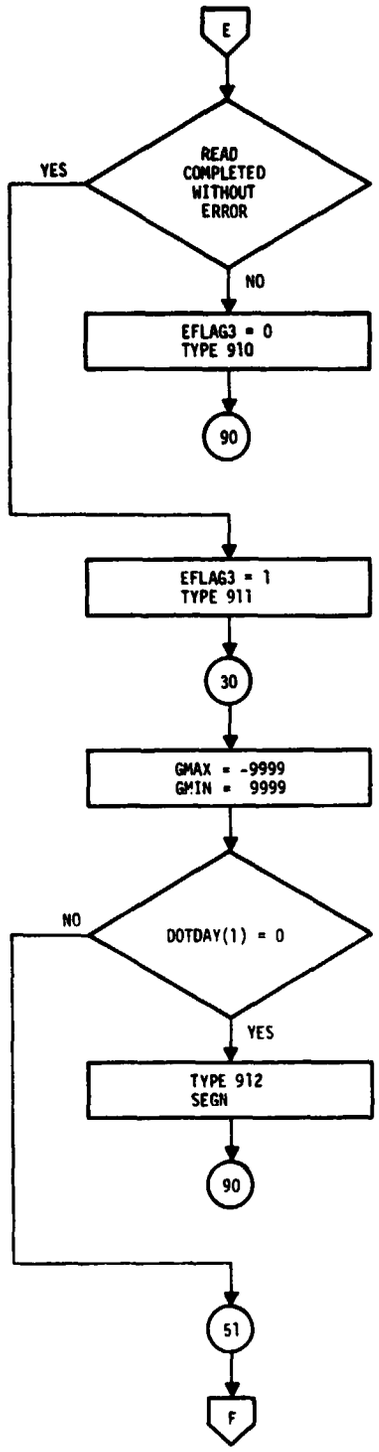
ORIGINAL PAGE IS
OF POOR QUALITY

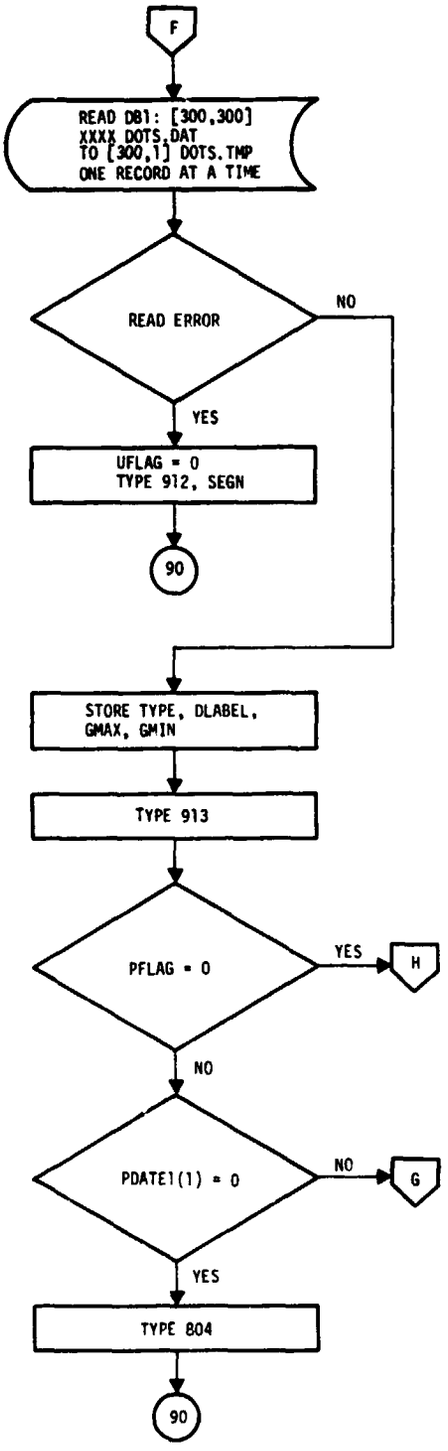
8. INITIATE SEGMENT ANALYSIS

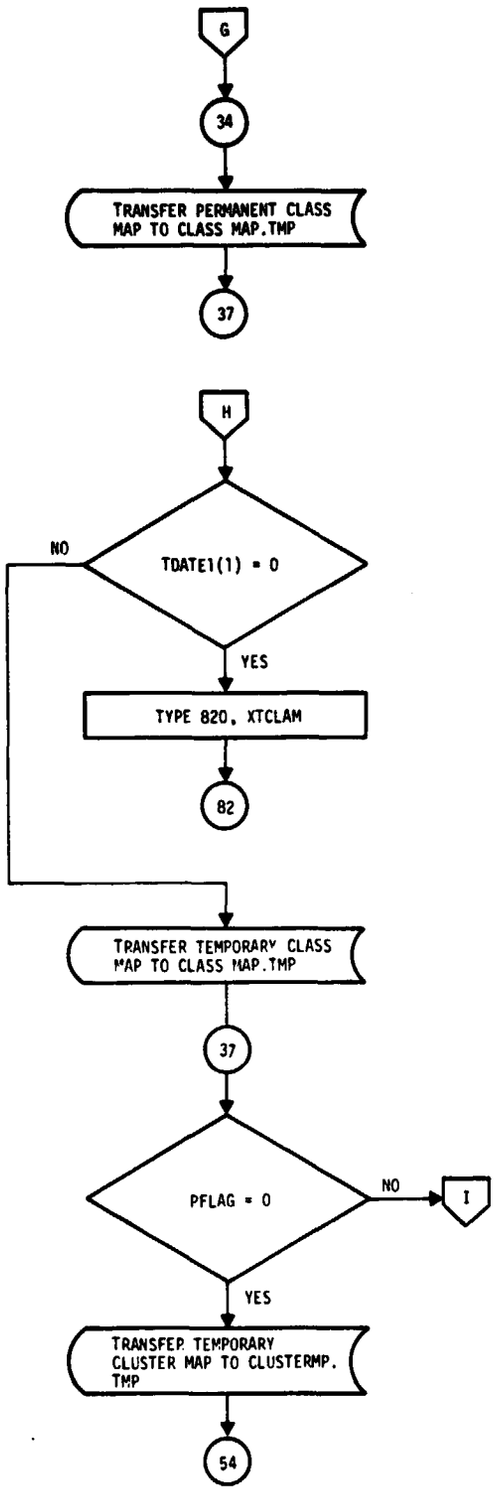


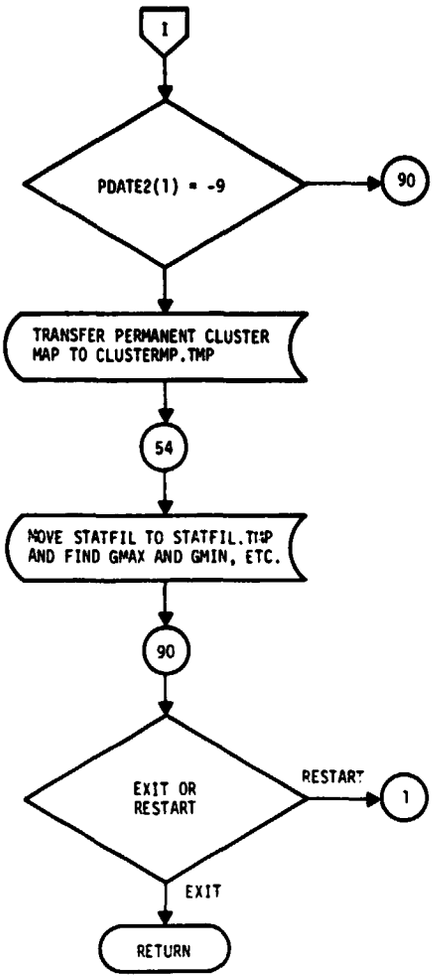

A
TYPE 903 SEGN
ACCEPT Y, N, X, OR B
KEYIN
NONE OF THE OTHERS
10
B
1
X
90
Y
B
N
TYPE 816
5
B
TYPE 815 ACCEPT CR, X, B
KEYIN
X
90
OTHER
31
B
5
C
40
RECLOC = PTR
READ C2 ISW = 2
41
CLEAR SCREEN AND WRITE SUMMARY REPORT
ISW
1
90
≠ 1
46
TYPE 914 ACCEPT KEYIN
KEYIN
X
90
R
10
NONE OF THE OTHERS
46
C OR B
60
CLEAR SCREEN AND PRINT TITLE
D


D
TYPE 906 ACCEPT ANALYST NAME
61
TYPE 907 ACCEPT KEYIN
KEYIN
B
60
X
90
P
65
T
62
TYPE 914
61
65
PFLAG = 1
66
62
TDATE1 =
TDATE2 =
TDATE3 =
YES
NO
TYPE 908
90
66
FLODAY(1)
≠0
= 0
EFLAG3 = 0
TYPE 909 SEGN
30
STORE DB1: [300, 300]
XXXX, FIELD. DAT
TO [300, 1] FIELD.TMP
E


E
READ COMPLETED WITHOUT ERROR
YES
NO
EFLAG3 = 0
TYPE 910
90
EFLAG3 = 1
TYPE 911
30
GMAX = -9999
GMIN = 9999
DOTDAY(1) = 0
NO
YES
TYPE 912
SEGN
90
51
F


F
READ DB1: [300,300]
XXXX DOTS.DAT
TO [300,1] DOTS.TMP
ONE RECORD AT A TIME
READ ERROR
NO
YES
UFLAG = 0
TYPE 912, SEGN
90
STORE TYPE, DLABEL,
GMAX, GMIN
TYPE 913
PFLAG = 0
YES
H
NO
PDATE1(1) = 0
NO
G
YES
TYPE 804
90


G
34
TRANSFER PERMANENT CLASS MAP TO CLASS MAP.TMP
37
H
NO
TDATE1(1) = 0
YES
TYPE 820, XTCLAM
82
TRANSFER TEMPORARY CLASS MAP TO CLASS MAP.TMP
37
PFLAG = 0
NO
I
YES
TRANSFER TEMPORARY CLUSTER MAP TO CLUSTERMP.TMP
54


I
PDATE2(1) = -9
90
TRANSFER PERMANENT CLUSTER MAP TO CLUSTERMP.TMP
54
MOVE STATFIL TO STATFIL.TMP AND FIND GMAX AND GMIN, ETC.
90
EXIT OR RESTART
RESTART
1
EXIT
RETURN

8.1 SUBROUTINE FTRNFR

```
FØRTRAN IV-PLUS V02-04          15:33:31    14-JUL-77          PAGE 14
INIT.FTN        /TR:BLØCKS/WR
        C       THIS SUBRØUTINE TRANSFER A FILE
0001          SUBRØUTINE FTRNFR(FILEIN,FILEØT,NR,KRW,MR,FLAG)
        C                   NR=NUMBER ØF RECØRDS IN THE FILE
        C                   LRW=LENTH ØF EACH RECØRD (WØRDS)
        C                   MR=MAXIMUM NUMBER ØF RECØRDS
0002          BYTE FILEIN(27),FILEØT(27)
0003          INTEGER ØUT,ARRAY(999),FLAG
0004          LRW=KRW/4
0005          JRW=LRW*2
0006          IN=7
0007          ØUT=9
0008          IPØINT=1
0009          ØPØINT=1
0010          ØPEN(UNIT=IN,NAME=FILEIN,TYPE='ØLD',ACCESS='DIRECT',
             1FØRM='UNFØRMATTED',ERR=2,BUFFERCØUNT=1,RECØRDSIZE=LRW,MAXREC=MR)
0011          ØPEN(UNIT=ØUT,NAME=FILEØT,TYPE='NEW',ACCESS='DIRECT',
             1ERR=3,RECØRDSIZE=LRW,MAXREC=MR)
0012    1     READ(IN'IPØINT,ERR=2)(ARRAY(I),I=1,JRW)
0013          WRITE(ØUT'PPØINT,ERR=3)(ARRAY(I),I=1,JRW)
0014          IF(IPØINT.GT.10)GØ TØ 6
0015    6     CØNTINUE
0016          IPØINT=IPØINT+1
0017          ØPØINT=IPØINT
0018          IF(IPØINT.LE.MR)GØ TØ 1
0019          FLAG=1
0020          GØ TØ 5
0021    2     TYPE 101
0022          GØ TØ 4
0023    3     TYPE 102
0024    4     FLAG=0
0025    5     CLØSE(UNIT=IN,DISPØSE='SAVE')
0026          CLØSE(UNIT=ØUT,DISPØSE='SAVE')
0027          RETURN
0028    101   FØRMAT(' READ ERRØR')
0029    102   FØRMAT(' WRITE ERRØR')
0030          END
```

8.1 SUBROUTINE FTRNFR

A flow chart for this subroutine is not available.

8.2 SUBROUTINE INTLZE

```
FØRTRAN IV-PLUS V02-04          15133:36    14-JUL-77          PAGE 16
INIT.FTN        /TR:BLØCKS'WR
0001            SUBRØUTINE I TL2E
0002            IMPLICIT INTEGER (A-Z)
0003            INCLUDE 'SY:[300,3]CAMSCØMØN.INC'
0004  *         INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0005  *         PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      *        1,MAXV=11,NDØTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *        2NØSPWD=6,NØDTWD=10
0006  *         EQUIVALENCE (C1,ACDATE),(C2,ISEG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0007  *         INTEGER C1(469),C2(256),C3(71),C4(348),C5(629)
      *  C*
0008  *         INTEGER ACDATE,SUBCAT,SUBPØP,CATKNT,CATTH
0009  *         BYTE CHNVEC,NØCHAN,NØSUB,DØTCAT,DØTCLU
0010  *         CØMMØN/CØM1/ACDATE(2,MAXACC),CHNVEC(MAXCHN,MAXACC),NØCHAN,NØSUB,
      *        1SUBCAT(MAXSUB),SUBPØP(MAXSUB),CATKNT(MAXCAT),CATTH(MAXCAT),NØDØ,
      *        2NØDU,NØTH,DØTCAT(NDØTS),DØTCLU(NDØTS)
      *  C*
0011  *         INTEGER ADATES,SUNAZ,ANALST,FLDDAY,DØTDAY,PDATE1,TDATE1
0012  *         INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RANDØM,GRID
0013  *         BYTE DELFLG,NØACQ,SØILGR,SUNEL,NSTART,NTYPE1,ALP,ALPØ
0014  *         BYTE PCTCT,PCTCTØ,VAR,VARØ,DLABEL,TYPE
0015  *         CØMMØN/CØM2/ISEG,DELFLG,NØACQ,ADATES(2,MAXACD),SØILGR(MAXACD),
      *        1SUNEL(MAXACD),SUNAZ(MAXACD),IMDATE(2),ANALST(5),FLDDAY(2),
      *        2DØTDAY(2),NSTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      *        3PDATE3(2),TDATE3(2),NØCAT,CATNAM(MAXCAT),ALP(MAXCAT),ALPØ,
      *        4          PCTCT(MAXCAT),PCTCTØ,VAR(MAXCAT),VARØ
      *  C*
0016  *         INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      *        1UFLAG4
0017  *         INTEGER PFLAG,DSKMNT
0018  *         CØMMØN/CØM3/PFLAG,DSKMNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
      *        1,UFLAG2,UFLAG3,UFLAG4,NEWLAB(MAXSUB)
      *  C*
0019  *         INTEGER TX1,TY1,TX2,TY2,ACDISP,G,B,DTWIND,DØTARY,GMIN,GMAX,FUL
0020  *         INTEGER SPWIND,CLAWND,CLUWND
0021  *         CØMMØN/CØM4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACDISP(2),I11(4),G(4),
      *        1B(4),DTWIND(5,NØDTWD),SPWIND(5,NØSPWD),IMWIND(4),NUMDØT,
      *        2DØTARY(NDØTS),GMIN,GMAX,FUL(2,7),CLAWND(8),CLUWND(8)
0022  *         CØMMØN/CØM5/DISKID,RANDØM(NDØTS),GRID(NDØTS),DLABEL(NDØTS),
      *        1TYPE(NDØTS),RECLØC
0023            DIMENSION RANDØM(NDØTS)
0024            INCLUDE 'SY:[300,3]RANDØM.DAT'
0025  *         DATA RANDØM/209,67,208,14,127,99,35,196,191,95,66,197,122,183,
      *        1114,51,80,206,30,90,172,199,110,56,17,44,135,174,48,121,188,184,2,
      *        2112,132,4,181,43,60,158,63,20,198,94,39,86,125,118,104,134,82,185,
      *        315,147,157,77,75,161,3,76,187,202,8,136,73,89,152,13,207,153,9,
      *        4175,33,32,24,162,131,42,18,49,203,36,1,155,96,87,193,171,50,19,70,
      *        5120,164,68,54,194,81,93,166,173,22,105,180,115,5,27,46,12,38,16,
      *        684,78,28,204,124,69,176,106,144,143,6,141,72,59,10,201,71,190,107,
      *        7130,40,88,146,102,31,182,137,62,177,138,167,163,51,34,129,151,186,
      *        8113,119,23,52,21,109,150,105,29,145,55,209,108,92,117,64,97,79,
      *        991,45,128,165,26,116,53,139,169,111,160,7,47,11,148,85,154,58,74,
      *        A192,169,123,98,178,140,205,101,65,103,57,126,179,142,83,156,25,
      *        B189,133,170,100,41,149,159,37/
0026            DØ 5 I=1,469
0027            C1(I)=0
0028      5     CØNTINUE
```

```
0029            DO 6 I=1,256
0030     6      C2(I)=0
0031            DO 7 I=1,71
0032     7      C3(I)=0
0033            DO 8 I=1,348
0034     8      C4(I)=0
0035            DO 9 I=1,629
0036     9      C5(I)=0
0037            DO 1 I=1,NDOTS
0038            RANDOM(I)=PANDOM(I)
0039              K=RANDOM(I)
0040              GRID(K)=1
0041            DOTARY(I)=0
0042     1      CONTINUE
0043            DO 4 I=1,469
0044     4      C1(I)=0
0045            SPWIND(1,1)=0
0046            SPWIND(2,1)=50
0047            SPWIND(3,1)=269
0048            SPWIND(4,1)=152
0049            SPWIND(5,1)=371
0050            SPWIND(1,2)=0
0051            SPWIND(2,2)=204
0052            SPWIND(3,2)=269
0053            SPWIND(4,2)=306
0054            SPWIND(5,2)=371
0055            SPWIND(1,3)=0
0056            SPWIND(2,3)=358
0057            SPWIND(3,3)=269
0058            SPWIND(4,3)=460
0059            SPWIND(5,3)=371
0060            SPWIND(1,4)=0
0061            SPWIND(2,4)=50
0062            SPWIND(3,4)=388
0063            SPWIND(4,4)=152
0064            SPWIND(5,4)=490
0065            SPWIND(1,5)=0
0066            SPWIND(2,5)=204
0067            SPWIND(3,5)=388
0068            SPWIND(4,5)=306
0069            SPWIND(5,5)=490
0070            SPWIND(1,6)=0
0071            SPWIND(2,6)=358
0072            SPWIND(3,6)=388
0073            SPWIND(4,6)=460
0074            SPWIND(5,6)=490
0075            DO 2 I=1,MAXSUB
0076            NEWLAB(I)=SURCAT(I)
0077     2       CONTINUE
0078            TX1=1
0079            TY1=1
0080            TX2=196
0081            TY2=117
0082            IX1=0
0083            IY1=20
0084            IX2=391
```

```
0085            IY2=253
0086            DO 3 I=1,5
0087            DO 3 J=1,NODTWD
0088            DTWIND(I,J)=0
0089     3      CONTINUE
0090            IMWIND(1)=0
0091            IMWIND(2)=20
0092            IMWIND(3)=391
0093            IMWIND(4)=253
0094            NUMDOT=0
0095            FUL(1,1)=0
0096            FUL(2,1)=128
0097            FUL(1,2)=0
0098            FUL(2,2)=128
0099            FUL(1,3)=0
0100            FUL(2,3)=125
0101            FUL(1,4)=0
0102            FUL(2,4)=64
0103            FUL(1,5)=0
0104            FUL(2,5)=100
0105            FUL(1,6)=0
0106            FUL(2,6)=120
0107            FUL(1,7)=-5
0108            FUL(2,7)=50
0109            EFLAG4=0
0110            UFLAG1=0
0111            UFLAG2=0
0112            UFLAG3=0
0113            UFLAG4=0
0114            RETURN
0115            END
```

8.2 SUBROUTINE INTLZE

A flow chart for this subroutine is not available.

9. IMAGE DISPLAY FUL013

```
HEWHTRAN IV-PLUS V02-04              13136112   31-AUG-77          PAGE 1
FUL013.FTN        /TR;HLOCKS/WR
0001                PROGRAM FUL013
0002                INCLUDE 'SYS:FUL01COM.COM'
      *  C          THIS VERSION OF FUL01COM.COM IS TAYLORED TO FUL013,AND FUL014,
      *  C          THE CAMS HYBRID VERSIONS. TO CHANGE TO THE RTE VERSIONS, SET
      *  C          NCMAT TO 64 AND CHANGE THESE COMMENTS.
0003  *             IMPLICIT INTEGER(A - Z)
0004  *               PARAMETER NCMAT=8
0005  *             REAL GB,RFAC,XZ,YZ
0006  *             BYTE FILE,MGL,MGF,LGL,LGF,HFL,HFG,CHL,CHP,WC,MXC,LUT
0007  *             COMMON /FCOM/FILE(33),FC(5),MGL,MGF,LGL,LGF,HFL,HFG,CHL(NCMAT),
      *            1 CHP(NCMAT),MXC(NCMAT,5),NF,F11,MTXTFG,TUNIT,TUN,FLN,FORM,
      *            2 TTX1,TTY1,TTX2,TTY2,IIX1,IIY1,IIX2,IIY2,IX(512),IY(512),
      *              3 TX(512),TY(512),MX,MY,NX,NCMAX,LGBC(5,3),LUN,GB(NCMAT+1,5)
      *              4 ,LUT(256,5),NCH,RFAC,XZ,YZ
0008  *              COMMON /HCOM/SS,SF,LS,LF, NRPDS,NDSPR,NCPR,NRPC,ANCL,
      *              1 MC,NS,NBIT,DPI,NCAR,SVD,PSIZ,PSKIP,MSIZ,CALP,CERR
0009  *              COMMON /FATAL/ZD,KR
0010  *              COMMON /IOLP/IO,LP
0011  *             INTEGER FCOM(2762+27*NCMAT/2),HCOM(20)
0012  *             EQUIVALENCE (FCOM(1),FILE(1))
0013  *             EQUIVALENCE ( HCOM(1),SS)
0014                  INCLUDE 'SYS:[300,3]CAMSCOM.INC'
0015  *               INCLUDE 'SYS:[300,3]CAMSPARAM.INC'
0016  *               PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      *              1,MAXV=11,NDATS=209,ILSKID=13,DSSKIP=10,MAXACD=6,MAXACC=4,
      *              2MAXSPW=60,NDDT=10
0017  *               EQUIVALENCE (C1,ACDATE),(C2,ISFG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0018  *               INTEGER C1(469),C2(256),C3(71),C4(348),C9(629)
      *    C*
0019  *               INTEGER ACDATE,SUBCAT,SUBPOP,CATNAM,CATTH
0020  *               BYTE CHNVEC,NOCHAN,NOSUB,DOTCAT,DOTCLU
0021  *               COMMON /COM1/ACDATE(2,MAXACC),CHNVEC(MAXCHN,MAXACC),NOCHAN,NOSUB,
      *              1SUBCAT(MAXSUB),SUBPOP(MAXSUB),CATNAM(MAXCAT),CATTH(MAXCAT),NODO,
      *              2NODU,NOTH,DOTCAT(NDATS),DOTCLU(NDOTS)
      *    C*
0022  *               INTEGER ADATES,SUBSZ,ANALST,FLDDAY,DOTDAY,PDATE1,TDATE1
0023  *               INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RANDOM,GRID
0024  *               BYTE SELFLG,NOACD,SPILGR,SUBSEL,NSTART,NTYPE1,ALP,ALP0
0025  *               BYTE PCTCT,PCTCT2,VAR,VAR2,CLASEL,TYPE
0026  *               COMMON /COM2/ISFG,SELFLG,NOACD,ADATES(2,MAXACD),SPILGR(MAXACD),
      *              1SUBSEL(MAXACD),SUBAV(MAXACD),TDATE(2),ANALST(5),FLDDAY(2),
      *              2DOTDAY(2),NSTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      *              3PDATE3(2),TDATE3(2),NPCAT,CATNAM(MAXCAT),ALP(MAXCAT),ALP0,
      *              4           PCTCT(MAXCAT),PCTCT2,VAR(MAXCAT),VAR2
      *    C*
0027  *               INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      *              1UFLAG4
0028  *               INTEGER PFLAG,DSKMNT
0029  *               COMMON/COM3/PFLAG,DSKMNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
      *              1,UFLAG2,UFLAG3,UFLAG4,SENL(MAXSUB)
      *    C*
0030  *               INTEGER TX1,TY1,TX2,TY2,ACDISP,G,B,DTWIND,DOTARY,GMIN,GMAX,FUL
0031  *               INTEGER SPWIND,CLAWND,CLUWND
0032  *               COMMON/COM4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACDISP(2),III(4),G(4),
      *              1B(4),DTWIND(5,NDOTSD),SPWIND(5,NDSPWD),IMWIND(4),NUMDOT,
      *              2DOTARY(NDOTS),GMIN,GMAX,FUL(2,7),CLAWND(8),CLUWND(8)
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
0033 *        CØMMØN/CØMS/DISKID,RANDØM(NDØTS),GRID(NDØTS),DLABEL(NDØTS),
     *       1TYPE(NDØTS),RECLØC
0034          BYTE DAET(9),TIM(8),CRF,LINE(74),SHFL,HREC(3060),BUF(3300)
0035          INTEGER DØTS(2,209)
0036          EQUIVALENCE (DØTS(1,1),BUF(1))
0037          EQUIVALENCE(BUF(13),HREC(1))
0038          EQUIVALENCE(BUFW,BUF(1))
0039          REAL GAIN,A2,A3
0040          BYTE UIC(9),EXT(5)
0041          BYTE DEV(5)
0042          INTEGER A1,J,TØ,S,FUL014(2),LUNN,LUTAB(10),
             1 BUFS2,BUFBIG,NCR,I,IF,ICH,TI,IP,SHFG,PRTY,E2F
0043          DATA DAET,TIM,SHFL,SHFG/17*' ','Y',0/
0044          DATA IP,LP/6,5/
0045          DATA S,LUNN/0,0/
0046          DATA FUL014/3RFUL,3R014/
0047          DATA BUFBIG,TI,PRTY,E2F/10000,4,0,0/
0048          DATA FILE(1),FILE(2),FILE(3),FILE(4),FILE(5)/'M','T','0','1',0/
0049          DATA UIC,EXT/'[','3','0','0',',','3','0','0','3',',','D','A',
              1 'T',0/
0050          DATA DEV/'D','B','0','2',':'/
0051          NF = 5
0052          DØ 6001 I = 1,4
0053          GB(NCMAT+1,I)=0
0054          LGRC(I,1)=I
0055          LGRC(I,2)=200
0056          LGRC(I,3)=0
0057          DØ 6001 J = 1,NCMAT
0058          MXC(J,I)=J
0059 6001     GB(J,I)=1.
0060          LGRC(4,2)=400
0061          LGRC(5,1)=0
0062          RWD = 0
0063          F11 = 1
0064          MTXTFG = 0
0065          TUNIT = 0
0066          TUN = 1
0067          FLN = 1
0068          FØRM = 1
0069          TTX1 = 1
0070          TTY1 = 1
0071          TTX2 = 512
0072          TTY2 = 512
0073          IIX1 = 0
0074          IIY1 = 0
0075          IIX2 = 511
0076          IIY2 = 511
0077          LGL = 'Y'
0078          LGF = 0
0079          HFL = 'N'
0080          HFG = 1
0081          MGL = 'N'
0082          MGF = 1
0083          PAGE = 1
0084          PASS=1
0085          DSETF = 0
```

```
0086                ATF = 1
0087                RFAC = 0.
0088                DØG = 0
0089                MY = 0
0090                MX = 0
0091                NX = 0
0092                LCDF='Y'
0093                LCDG=0
0094                J=1
0095                DØ 3001 I = 1,MAXACD
0096                IF(ADATES(1,I)-ADATES(1,J))3001,3002,3003
0097     3002       IF(ADATES(2,I).LE.ADATES(2,J))GØ TØ 3001
0098     3003       J=I
0099     3001       CØNTINUE
0100                ACKN=J
0101     1          IF(DSETF.NE.0)CALL DUSET(1,F11,1)
0102                DSETF=0
0103                CALL CSGRPH(10,PAGE,'IMAGE DISPLAY 1,0 5/2/77',24)
0104     3000       IF(TI.EQ.12)WRITE(12,4000)ISEG,(J,(ADATES(I,J),I=1,2),
                    1 J=1,MAXACD),ACKN
0105     4000       FØRMAT(' FØR SEGMENT ',I4,' SELECT ØNE ØF THE FØLLØWING'/
                    1 18X,'ACQUISITION DATES FØR IMAGE DISPLAY'/<MAXACD-1>(18X,'(',
                    2 I1,')  ',I2,I3/),'3',17X,'(',I1,')  ',I2,I3,10X,I1,'  > ')
0106                CALL LIN(IP,LINE,TI,IF)
0107                GØ TØ (9401,1,1,3000,3004,3005),IF
0108     3005       IP = 0
0109                CALL INTFF(IP,LINE,74,I)
0110                IF(I.LE.0.ØR.I.GT.MAXACD)GØ TØ 3000
0111                ACKN = I
0112     3004       J = 24
0113                K = ISEG
0114                L = ADATES(1,ACKN)
0115                M = ADATES(2,ACKN)
0116                DØ 3006 I = 1,3
0117                N = M-(M/10)*10
0118                M = M/10
0119                J = J-1
0120     3006       FILE(J)=N+"60
0121                DØ 3007 I = 1,2
0122                N = L-(L/10)*10
0123                L = L/10
0124                J = J-1
0125     3007       FILE(J)=N+"60
0126                DØ 3008 I = 1,4
0127                N = K-(K/10)*10
0128                K = K /10
0129                J = J-1
0130     3008       FILE(J)=N+"60
0131                DØ 3030 I = 1,9
0132     3030       FILE(I+5)=UIC(I)
0133                DØ 3037 I = 1,5
0134     3037       FILE(I)=DEV(I)
0135                DØ 3031 I = 1,5
0136     3031       FILE(I+23)=EXT(I)
           C        WRITE(10,4500)(FILE(I),I=1,23)
           C4500    FØRMAT(' ',32A1)
```

BEST AVAILABLE COPY
due to poor quality

```
0137              NF=24
0138              SHFG = 1
0139              PRTY = 0
0140              F11 = 0
0141              FØRM = 1
0142    3009      IF(TI.EQ.10)WRITE(10,4001)LCDF
0143    4001      FØRMAT('$LACIE DEFAULTS ? (Y)ES/(N)O  ',A1,'  > ')
0144              CALL LIN(I0,LINE,TI,IF)
0145              G0 T0 (9401,3000,1,3009,3010,3011),IF
0146    3010      LINE(1)=LCDF
0147    3011      LCDG = 1
0148              IF(LINE(1).EQ.'N') G0 T0 502
0149              IF(LINE(1).NE.'Y') G0 T0 3009
0150              LCDG = 0
0151              NCMAX=4
0152              ITX1 = 1
0153              ITY1 = 1
0154              ITX2 = 196
0155              ITY2 = 117
0156              IIX1 = 0
0157              IIY1 = 20
0158              IIX2 = 391
0159              IIY2 = 253
0160              D0 3023 I = 1,4
0161    3023      LGRC(I,1)=I
0162              LGRC(5,1)=0
0163    502       IF(DSETF.EQ.1)CALL DUSET(1,F11,1)
0164              CALL DSET(1,F11,FILE,MTXTFG,TUNIT,FF,1)
0165              DSETF = 1
0166              CALL HREAD(1,HREC,BUF,3060,FØRM,F11,PRTY,EØF,SHFG)
0167              IF(CERR.EQ.0) GØTØ 503
0168    504     WRITE(10,7000)CERR
0169    7000    FØRMAT(' CATASTR0PHIC ERR0R NUMBER ',I5/)
0170            G0 T0 17
0171    505       CALL DUSET(1,F11,1)
0172              DSETF = 0
0173              G0 T0 3009
0174    503       CØNTINUE
0175              IF(LCDG.EQ.0) G0 T0 902
0176              IF(TI.EQ.12) WRITE(12,8050)NC,SS,SE,LS
0177    8050    FØRMAT(' THE DATA HAS ',I2,' CHANNELS'/' THE START PIXEL IS  'I5/
               1 ' THE END PIXEL IS ',I5/' THE START LINE NUMBER IS  'I5/)
0178              IF(PASS.NE.1) G0 T0 6
0179              PASS=PASS+1
0180              ITX1 = SS
0181              ITX2 = SE
0182              ITY1 = LS
0183              ITY2=LS+511
0184              IF(SE-SS.NE.195) G0 T0 6
0185              ITY2=LS+116
0186              IIX1 = 0
0187              IIY1 = 20
0188              IIX2 = 391
0189              IIY2 = 253
0190    6       IF(TI.EQ.10)WRITE(10,8006)ITX1,ITY1
0191    8006    FØRMAT('$UPPER LEFT C0ORDINATES 0F DATA BASE IMAGE = ',
```

```
                1 I5,' ',', ',I5,'  > ')
0192              CALL LIN(I0,LINE,TI,IF)
0193              G0 T0 (9401,505,1,6,7,106),IF
0194    106       C0NTINUE
0195            IP = 0
0196            CALL INTFF(IP,LINE,73,TTX1)
0197            CALL INTFF(IP,LINE,73,TTY1)
0198            IF(1.GT.TTX1.0R.TTX1.GT.196)G0 T0 601
0199              IF(1.LE.TTY1.AND.TTY1.LE.117) G3 T0 7
0200    601     TI = I0
0201            WRITE(I0,7001)LINE
0202    7001    F0RMAT(' IMPR0PER INPUT LINE F0LL0WS'/' ',73A1)
0203            G0 T0 6
0204    7       IF(TI.EQ.I0)WRITE(I0,8007)TTX2,TTY2
0205    8007    F0RMAT('$LOWER RIGHT C00RDINATES 0F DATA BASE IMAGE = ',
                1 I5,' ',', ',I5,'  > ')
0206              CALL LIN(I0,LINE,TI,IF)
0207              G0 T0 (9401,6,1,7,703,107),IF
0208    107       C0NTINUE
0209            IP = 0
0210            CALL INTFF(IP,LINE,73,TTX2)
0211            CALL INTFF(IP,LINE,73,TTY2)
0212            IF(1.GE.TTX2.0R.TTY2.GT.196) G0 T0 701
0213              IF(1.LE.TTY2.AND.TTY2.LE.117) G0 T0 702
0214    701     TI = I0
0215            WRITE(I0,7001)LINE
0216            G0 T0 7
0217    702     IF(TTX1.EQ.TTX2.0R.TTY1.EQ.TTY2) G0 T0 701
0218    703       BUFSZ = BUFBIG-RSIZ
0219              IF(RSIZ.LT.HSIZ)BUFSZ=BUFBIG-HSIZ
0220            NCMAX = BUFSZ/(1+IABS(TTX1-TTX2))
0221              IF(NCMAX.LE.NCMAT) G0 T0 704
0222              NCMAX = NCMAT
0223              BUFSZ=NCMAT*(1+IABS(TTX1-TTX2))
0224    704       C0NTINUE
0225    8       IF(TI.EQ.I0)WRITE(I0,8008)IIX1,IIY1
0226    8008    F0RMAT('$UPPER LEFT C00RDINATES 0F SCREEN DISPLAY = '
                1 I5,' ',', ',I5,'  > ')
0227              CALL LIN(I0,LINE,TI,IF)
0228              G0 T0 (9401,7,1,8,9,108),IF
0229    108       C0NTINUE
0230            IP = 0
0231            CALL INTFF(IP,LINE,73,IIX1)
0232            CALL INTFF(IP,LINE,73,IIY1)
0233            IF(0.GT.IIX1.0R.IIX1.GT.511) G0 T0 801
0234            IF(0.LE.IIY1.AND.IIY1.LE.511) G0 T0 9
0235    801     TI = I0
0236            WRITE(I0,7001)LINE
0237            G0 T0 8
0238    9       IF(TI.EQ.I0)WRITE(I0,8009)IIX2,IIY2
0239    8009    F0RMAT('$LOWER RIGHT C00RDINATES 0F SCREEN DIPLAY = ',
                1 I5,' ',', ',I5,'  > ')
0240              CALL LIN(I0,LINE,TI,IF)
0241              G0 T0 (9401,8,1,9,902,109),IF
0242    109       C0NTINUE
0243            IP = 0
```

```
0244            CALL INTFF(IP,LINE,73,IIX2)
0245            CALL INTFF(IP,LINE,73,IIY2)
0246            IF(0.GT.IIX2.ØR.IIX2.GT.511) GØ TØ 901
0247            IF(0.GT.IIY2.ØR.IIY2.GT.511) GØ TØ 901
0248            IF(IIX1.EQ.IIX2) GØ TØ901
0249            IF(IIY1.NE.IIY2) GØ TØ 902
0250     901    TI = IØ
0251            WRITE(IØ,7001)LINE
0252            GØ TØ 9
0253     902    CALL ZØØØØM(IIX1,IIY1,IIX2,IIY2,TTX1,TTY1,TTX2,TTY2,IX,IY,TX,TY,
               1 XZ,YZ,MX,MY,NX)
0254            IF(ZØ.EQ.0) GØ TØ 10
0255            WRITE(IØ,7002)
0256     7002   FØRMAT(' ZØØØØM ERRØR - BAD IMAGERY/CRT PØSITIØNS.')
0257            TI = IØ
0258            GØ TØ 9
0259     10       CØNTINUE
0260              IF(LCDG.EQ.0)GØ TØ 1206
0261              MGL = 'N'
0262              MGF = 1
0263     12       CØNTINUE
0264     3017     IF(TI.EQ.IØ)WRITE(IØ,4003)
0265     4003     FØRMAT('$GAIN AND BIAS FRØM HEADER ? (Y)ES/(N)Ø  > ')
0266              CALL LIN(IØ,LINE,TI,IF)
0267              GØ TØ (9401,9,1,3017,3017,3018),IF
0268     3018     IF(LINE(1).EQ.'Y')GØ TØ 1206
0269              IF(LINE(1).NE.'N') GØ TØ 3017
0270              LGL = 'N'
0271              LGF = 0
0272              I = 0
0273     1203   I = I + 1
0274            IF(I.LE.4) GØ TØ 1204
0275            GØ TØ 2001
0276     1204   A1 = LGBC(I,1)
0277            A3 = FLØAT(LGBC(I,3))/100.
0278            A2=FLØAT(LGBC(I,2))/100.
0279            IF(TI.EQ.IØ)WRITE(IØ,8120)I,A1,A2,A3
0280     8120   FØRMAT(' ENTER IMAGERY CHANNEL, GAIN, BIAS FØR CRT CHANNEL ',I1/
               1 '$',I2,' ,',F7.2,' ,',F7.2,' : ')
0281              CALL LIN(IØ,LINE,TI,IF)
0282              GØ TØ (9401,12,1,12,1203,113),IF
0283     113      CØNTINUE
0284            IP = 0
0285            CALL INTFF(IP,LINE,73,J)
0286            IF(0.GT.J.ØR.J.GT.NC) GØ TØ 1205
0287            LGBC(I,1)=J
0288            IF(J.EQ.0) GØ TØ 1203
0289            CALL FFFPI(IP,LINE,73,GAIN)
0290              GAIN = GAIN * 100.
0291              IF(GAIN.GT.0.)GAIN=GAIN+.5
0292              IF(GAIN.LT.0.)GAIN=GAIN-.5
0293              LGBC(I,2)=GAIN
0294            CALL FFFPI(IP,LINE,73,A3)
0295              A3 = A3*100.
0296              IF(A3.GT.0.)A3=A3+.5
0297              IF(A3.LT.0.)A3=A3-.5
```

```
FUL013.FTN      /TR:BLØCKS/WR
0298              LGBC(I,3)=A3
0299            GØ TØ 1203
0300     1205   IF(J.LT.0) WRITE(I0,8121 )
0301     8121   FØRMAT(' CHANNEL NUMBER MUST BE PØSITIVE.')
0302            WRITE(I0,8122)J
0303     8122   FØRMAT(' CHANNEL ',I10,' NØT IN IMAGERY.')
0304            TI = I0
0305              WRITE(I0,7001)LINE
0306            GØ TØ 12
0307     1206     LGL=LINE(1)
0308              LGF=0
0309     3013     HFL='N'
0310              HFG=1
0311     1401   CØNTINUE
0312     15       CØNTINUE
0313              IF(LCDG.EQ.0)GØ TØ 3016
0314     3015     IF(TI.EQ.I0)WRITE(I0,4002)
0315     4002     FØRMAT('$DEFAULT CHANNEL ASSIGNMENT ? (Y)ES/(N)Ø   > ')
0316              CALL LIN(I0,LINE,TI,IF)
0317              GØ TØ (9401,12,1,3015,3015,3036),IF
0318     3036     IF(LINE(1).EQ.'N') GØ TØ 3014
0319              IF(LINE(1).NE.'Y') GØ TØ 3015
0320              DØ 3034 I = 1,4
0321     3034     LGBC(I,1)=I
0322              LGBC(5,1)=0
0323              GØ TØ 3016
0324     3014     I = 0
0325     1501   I = I + 1
0326            IF(I.LT.4) GØ TØ 1502
0327     3016     CALL GABI(HRE C)
0328              LGBC(5,1)=0
0329            GØ TØ 2002
0330     1502   IF(TI.EQ.I0)WRITE(I0,6015)I,LGBC(I,1)
0331     8015   FØRMAT('$ENTER IMAGERY CHANNEL FØR DISPLAY CHANNEL ',I1,' = '
                1 ,I5,'  > ')
0332              CALL LIN(I0,LINE,TI,IF)
0333              GØ TØ (9401,3015,1,1502,1501,115),IF
0334     115      CØNTINUE
0335            IP = 0
0336            CALL INTEF(IP,LINE,73,J)
0337            IF(J.LT.0.ØR.J.GT.NC) GØ TØ 1503
0338            LGBC(I,1)=J
0339            GØ TØ 1501
0340     1503   WRITE(I0,8121)
0341            WRITE(I0,8122)J
0342            TI = I0
0343            GØ TØ 12
0344     2000     CØNTINUE
0345     2001     CØNTINUE
0346     2002   CALL TCHLST(NCR)
0347              IF(LGF.EQ.0)CALL CØMLUT
0348     18       CØNTINUE
0349     3019     REAC = 0.
0350     2007     CØNTINUE
0351              TØ = I0
0352              IF(TI.EQ.I0)CALL CSGDPH(I0,PAGE,
```

```
                1 'IMAGE DISPLAY 1.0 5/2/77',24)
0353              IF(TI.NE.6)T0 = LP
0354                WRITE(T0,4004)(DATES(I,ACKN),I=1,2)
0355      4004      FØRMAT(' ACQUISITI2N DATE = ',I2,I3)
0356                WRITE(T0,4005)TTY1,TTY1
0357                WRITE(T0,4006)TTX2,TTY2
0358                WRITE(T0,4007)IIX1,IIY1
0359                WRITE(T0,4008)IIX2,IIY2
0360      4005      FØRMAT(' UPPER LEFT C0ØRDINATES ØF DATA BASE IMAGE = ',
                1 I5,',',I5)
0361      4006      FØRMAT(' LØWER RIGHT C0ØRDINATES ØF DATA BASE IMAGE = ',
                1 I5,',',I5)
0362      4007      FØRMAT(' UPPER LEFT C0ØRDINATES ØF SCREEN DISPLAY = ',
                1 I5,',',I5)
0363      4008      FØRMAT(' LØWER RIGHT C0ØRDINATES ØF SCREEN DISPLAY = ',
                1 I5,',',I5)
0364      3021      WRITE(T0,2110)
0365      2110    FØRMAT(' DISPLAY-CH IMAGERY-CH        GAIN          BIAS')
0366              D0 2000 I = 1,4
0367              GAIN = FL0AT(LG0C(I,2))/100.
0368                A3 = FL0AT(LG0C(I,3))/100.
0369      2000    WRITE(T0,2111)I,LG0C(I,1),GAIN,A3
0370      2111    FØRMAT(6X,I1,6X,I2,4X,F10.2,2X,F10.2)
0371      10      C0NTINUE
0372              CLØSE( UNIT = T0 )
0373      1606    C0NTINUE
0374              IF(TI.EQ.12)WRITE(12,8016)
0375      8016    FØRMAT('$PR0CEED (Y/N) : ')
0376                CALL LIN(I2,LINE,TI,IF)
0377                GØ T0 (9401,3035,1,1606,1606,116),IF
0378      3035      IF(LC0G.EQ.0)GØ T0 3000
0379                GØ T0 12
0380      116     IF(LINE(1).EQ.'N') GØ T0 1607
0381              IF(LINE(1).EQ.'Y') GØ T0 1601
0382              TI = 12
0383                WRITE(13,7001)LINE
0384              GØ T0 1606
0385      1607      CALL DUSET(1,F11,1)
0386                DSETF = 0
0387                GØ T0 1
0388      1601    ØPEN(UNIT=2,NAME='F300,1]F:L2I.DAT',TYPE='UNKN0WN',FØRM=
                1 'UNF2RMATTED',ERR = 1602)
0389              GØ T0 1603
0390      1602    WRITE(12,8160)
0391      8160    FØRMAT(' ØPEN ERRØR ØN FILE FUL2I.DAT'/)
0392              TI = 1
0393              GØ T0 17
0394      1603    WRITE(2)FC2M,HC0M
0395              CLØSE(UNIT=2,DISPØSE='SAVE')
0396              CALL DUSET(1,F11,1)
0397                DSETF = 0
0398              CALL CLREF(53)
0399                TX1 = TTX1
0400                TX2 = TTX2
0401                TY1 = TTY1
0402                TY2 = TTY2
```

```
FUL013.FTN      /IR:BL0CKS/WR
0403            IX1 = IIX1
0404            IX2 = IIX2
0405            IY1 = IIY1
0406            IY2 = IIY2
0407            D0 3022 I = 1,MAXCHN
0408            G(I)=LGBC(I,2)
0409            B(I)=LGBC(I,3)
0410    3022    II1(I)=LGBC(I,1)
0411            ACDISP(1)=ADATES(1,ACKN)
0412            ACDISP(2)=ADATES(2,ACKN)
0413            0PEN(UNIT=2,NAME='[300,1]D0TGXY.TMP',
                1 TYPE='UNKN0WN',F0RM='UNF2RMATTED')
0414            D0 3038 I = 1,ND0TS
0415            D0TS(1,I)=0
0416    3038    D0TS(2,I)=0
0417            I1 = TTX1
0418            I2 = TTX2
0419            I3 = TTY1
0420            I4 = TTY2
0421            IF(I1.LE.I2)G0 T0 3040
0422            I = I1
0423            I1 = I2
0424            I1 = 1
0425    3040    IF(I3.LE.I4)G0 T0 3041
0426            I = I3
0427            I3 = I4
0428            I4 = 1
0429    3041    C0NTINUE
0430            I1 = (I1+9)/10
0431            I2 = I2/10
0432            I3 = (I3+9)/10
0433            I4 = I4/10
0434            IF(I1.GT.I2.0R.I3.GT.I4) G0 T0 3042
0435            D0 3039 I = I3,I4
0436            D0 3039 J = I1,I2
0437            K = (I-1)*19+J
0438            D0TS(1,K)=(J*10-TTX1)*XZ+IX1+.5
0439    3039    D0TS(2,K)=(I*10-TTY1)*YZ+IY1+.5
0440            WRITE(2)D0TS
0441    3042    CL0SE(UNIT=2)
0442          CALL REQUES(FUL014)
0443          CALL WAITFR(53)
0444    17    IF(TI.EQ.I0)WRITE(IW,8017)
0445    8017  F0RMAT('$E(X)IT, 0R (R)ESTART  : ')
0446            CALL LIN(I0,LINE,TI,IF)
0447            G0 T0 (9401,1606,1,17,17,117),IF
0448    117   IF(LINE(1).EQ.'R') G0 T0 1
0449          TI = I0
0450          G0 T0 17
0451    9401    IF(0SETF.EQ.1)CALL DUSET(1,F11,0)
0452            I = 50
0453    9400  CALL SETEF(I)
0454          CALL EXIT
0455          END
```

9.1 SUBROUTINE CRUNCH

```
HFORTRAN IV-PLUS V02-04          13138104    31-AUG-77          PAGE 1
CRUNCH.FTN        /TR:3LOCKS/WR
0001              SUBROUTINE CRUNCH
0002                INCLUDE 'SY:FUL0IC0M.C0M'
      *  C          THIS VERSION OF FUL0IC0M.C0M IS TAYLORED TO FUL013,AND FUL014.
      *  C          THE CAMS HYBRID VERSIONS. TO CHANGE TO THE RTE VERSIONS, SET
      *  C          NCMAT TO 64 AND CHANGE THESE COMMENTS.
0003  *            IMPLICIT INTEGER(A - Z)
0004  *              PARAMETER NCMAT=6
0005  *            REAL GB,RFAC,XF,YF
0006  *            BYTE FILE,MGL,MGF,LGL,LGF,HFL,HFG,CHL,CHP,WC,MXC,LUT
0007  *            COMMON /FCOM/FILE(33),WC(5),MGL,MGF,LGL,LGF,HFL,HFG,CHL(NCMAT),
      *           1 CHP(NCMAT),MXC(NCMAT,5),NF,F11,MTXTFG,TUNIT,TUN,FLN,FORM,
      *           2 TTX1,TTY1,TTX2,TTY2,IIX1,IIY1,IIX2,IIY2,IX(512),IY(512),
      *             3 TX(512),TY(512),MX,MY,NX,NCMAX,LGBC(5,3),LUN,GB(NCMAT+1,5)
      *             4 ,LUT(256,5),NCM,RFAC,XF,YF
0008  *             COMMON/HCOM/SS,SE,LS,LE,NRPOS,NDSPR,NCPR,NRPC,ANCL,
      *             1 AC,NS,NBIT,DDI,NCAR,SVD,RSIZ,PSKIP,HSIZ,CALP,CERR
0009  *             COMMON /FATAL/ZZ,RR
0010  *             COMMON /IOLP/IA,LP
0011  *            INTEGER FCOM(2762+27*NCMAT/2),HCOM(20)
0012  *            EQUIVALENCE (FCOM(1),FILE(1))
0013  *            EQUIVALENCE( HCOM(1),SS)
0014              DO 1 I = 1,5
0015              K = 1
0016              DO 2 J = 1,NC
0017              IF(MXC(J,I),EQ,0) GO TO  2
0018              MXC(K,I)=MXC(J,I)
0019              GB(K,I)=GB(J,I)
0020              K = K + 1
0021        2     CONTINUE
0022              DO 3 J = K,NCMAT
0023              MXC(J,I)=0
0024        3     GB(J,I)=0
0025        1     CONTINUE
0026              RETURN
0027              END
```

9.2 SUBROUTINE GABI

```
HFORTRAN IV-PLUS V02-04          13138131    31-AUG-77          PAGE 1
GABI.FTN        /TR:PLOCKS/WR
0001          SUBROUTINE GABI(HR)
0002            INCLUDE 'SYIFUL0ICOM.COM'
      *  C      THIS VERSION OF FUL0IC0M.C0M IS TAYLORED TO FUL013,AND FUL014,
      *  C      THE CAMS HYBRID VERSIONS. TO CHANGE TO THE RTE VERSIONS, SET
      *  C      NCMAT TO 04 AND CHANGE THESE COMMENTS.
0003  *         IMPLICIT INTEGER(A - Z)
0004  *           PARAMETER NCMAT=5
0005  *         REAL GR,RFAC,XZ,YZ
0006  *         BYTE FILE,MGL,MGF,LGL,LGF,MFL,MFG,CHL,CHP,WC,MXC,LUT
0007  *         COMMON /FCOM/FILE(33),FC(5),MGL,MGF,LGL,LGF,MFL,MFG,CHL(NCMAT),
      *        1 CHP(NCMAT),MXC(NCMAT,5),NF,F11,MIXTFG,TUNIT,TUN,FLN,FORM,
      *        2 TTX1,TTY1,TTX2,TTY2,IIX1,IIY1,IIX2,IIY2,IX(512),IY(512),
      *          3 TX(512),TY(512),MX,MY,PX,NCMAX,LGBC(5,3),LUN,GB(NCMAT+1,5)
      *          4 ,LUT(256,5),NCR,RFAC,XZ,YZ
0008  *           COMMON/HCOM/SS,SR,LS,LE,NRPDS,NDSPR,NCPR,NRPC,ANCL,
      *          1 NC,NS,NBIT,DBI,PCAP,SVC,RS12,PSKIP,HS12,CALP,CERR
0009  *           COMMON /FATAL/FP,RP
0010  *           COMMON /IOLP/IO,LP
0011  *         INTEGER FCOM(2762+27*NCMAT/2),HCOM(20)
0012  *         EQUIVALENCE (FCOM(1),FILE(1))
0013  *         EQUIVALENCE( HCOM(1),SS)
0014              BYTE GR(1)
0015              REAL    R
0016            IF(FCOM.NE.1) RETURN
0017            DO 1 I = 1,5
0018            IF(LGBC(I,1),EQ,0) GO TO 1
0019            J = 4*LGBC(I,1)+2641
0020              K = FFUNC(HR(J))*256+FFUNC(HR(J+1))
0021              L = IAND(K,32767)
0022              IF(K.NE.L)L=-L
0023              LGBC(I,2)=L*10
0024              R=L
0025              R=R/10
0026              K=FFUNC(HR(J-2))*256+FFUNC(HR(J+1))
0027              L=IAND(K,32767)
0028              IF(K.NE.L)L=-L
0029              LGBC(I,3)=FLOAT(L)*10/R+.5
0030        1     CONTINUE
0031            IF(MFG,EQ,0) CALL HBCUTT
0032            RETURN
0033            END
```

ORIGINAL PAGE IS
POOR QUALITY

9.3 SUBROUTINE COMLUT

```
HP FORTRAN IV-PLUS V02-04          13138:27   31-AUG-77          PAGE 1
COMLUT.FTN        /TR:BLOCKS/WR
0001              SUBROUTINE COMLUT
0002                INCLUDE 'SY:FUL013COM.COM'
      *  C          THIS VERSION OF FUL013COM.COM IS TAYLORED TO FUL013,AND FUL014.
      *  C          THE CAMS HYBRID VERSIONS. TO CHANGE TO THE RTE VERSIONS, SET
      *  C          NCMAT TO 64 AND CHANGE THESE COMMENTS.
0003  *             IMPLICIT INTEGER(A - Z)
0004  *               PARAMETER NCMAT=4
0005  *             REAL GR,RFAC,XZ,YZ
0006  *             BYTE FILE,MGL,MGF,LGL,LGF,HFL,HFG,CHL,CHP,MC,MXC,LUT
0007  *             COMMON /FCOM/FILE(33),MC(5),MGL,MGF,LGL,LGF,HFL,HFG,CHL(NCMAT),
      *            1 CHP(NCMAT),MXC(NCMAT,5),GF,F11,NTXTFG,TUNIT,TUN,FLN,FORM,
      *            2 TTX1,TTY1,TTX2,TTY2,IIX1,IIY1,IIX2,IIY2,IX(512),IY(512),
      *              3 TX(512),TY(512),MX,MY,NX,NCMAX,LGBC(5,3),LUN,GR(NCMAT+1,5)
      *              4 ,LUT(256,5),NCR,RFAC,XZ,YZ
0008  *               COMMON /HCOM/SS,SE,LS,LE,NRPDS,NDSPR,NCPR,NRPC,ANCL,
      *              1 MC,NS,NPRIT,DPI,NCAR,SVD,RSIZ,PSKIP,HSIZ,CALP,CERR
0009  *               COMMON /FATAL/FC,HR
0010  *               COMMON /IOLP/IT,LP
0011  *             INTEGER FCOM(276+27*NCMAT/2),HCOM(20)
0012  *             EQUIVALENCE (FCOM(1),FILE(1))
0013  *             EQUIVALENCE( HCOM(1),SS)
0014               REAL A,B
0015               DO 1 I = 1,5
0016               IF(LGBC(I,1),EQ,0) GO TO 1
0017               A = FLOAT(LGBC(I,2))/100.
0018               B = FLOAT(LGBC(I,3))/100.
0019               DO 1 J = 1,256
0020               K=A*((J-1)+B)
0021                 IF(K.GT.255)K=255
0022                 IF(K.LT.0)K=0
0023                 LUT(J,I)=K
0024          1    CONTINUE
0025               RETURN
0026               END
```

9.4 SUBROUTINE TCHLST

```
HP FORTRAN IV-PLUS V02-04          13139124    31-AUG-77          PAGE 1
TCHLST.FTN        /TP:BLOCKS/NR
0001              SUBROUTINE TCHLST(NCRR)
0002                INCLUDE 'SYS:FUL013COM.COM'
      *  C          THIS VERSION OF FUL013COM.COM IS TAYLORED TO FUL013,AND FUL014,
      *  C          THE CAMS HYBRID VERSIONS. TO CHANGE TO THE RTE VERSIONS, SET
      *  C          NCMAT TO 64 AND CHANGE THESE COMMENTS.
0003  *             IMPLICIT INTEGER(A - Z)
0004  *               PARAMETER NCMAT=4
0005  *             REAL CR,RFAC,XZ,YZ
0006  *             BYTE FILE,MGL,MGF,LGL,LGF,MFL,MFG,CHL,CHP,NC,MXC,LUT
0007  *             COMMON /FCOM/FILE(33),NC(5),MGL,MGF,LGL,LGF,MFL,MFG,CHL(NCMAT),
      *            1 CHP(NCMAT),MXC(NCMAT,5),NF,F11,MTXTFG,TUNIT,TUN,FLN,FORM,
      *            2 TTX1,TTY1,TTX2,TTY2,IIX1,IIY1,IIX2,IIY2,IX(512),IY(512),
      *              3 TX(512),TY(512),NX,MY,MX,NCMAX,LGBC(5,3),LUN,GR(NCMAT+1,5)
      *              4 ,LUT(256,5),NC4,RFAC,XZ,YZ
0008  *               COMMON /HCOM/SS,SF,LS,LE,NRPDS,NCSPR,NCPR,NRPC,ANCL,
      *              1 NC,NS,NBIT,DCI,NCAR,SVD,ESIZ,PSKIP,HSIZ,CALP,CERR
0009  *               COMMON /EATAL/ZZ,RR
0010  *               COMMON /IELP/IZ,IF
0011  *             INTEGER FCAL(2762+27*NCMAT/2),HCOM(20)
0012  *             EQUIVALENCE (FCOM(1),FILE(1))
0013  *             EQUIVALENCE( HCOM(1),SS)
0014                DO 1 I = 1,NCMAT
0015          1     CHP(I)=0
0016                DO 2 I = 1,5
0017                NC(I)=0
0018                  IF(LFG.EQ.1) GO TO 3
0019                IF(LCHC(I,1),EQ,0)GO TO 3
0020                NC(I)=1
0021                J = LGBC(I,1)
0022                CHP(J)=J
0023          3       IF( GF.EQ.1) GO TO 2
0024                  DO 4 J = 1,NCMAT
0025                IF(MXC(J,I),EQ,0) GO TO 2
0026                NC(I)=1
0027          4     CHP(J)=J
0028          2     CONTINUE
0029                K = 1
0030                DO 5 I = 1,NCMAT
0031                IF(CHP(I),EQ,0) GO TO 5
0032                CHL(K)=I
0033                CHP(I)=K
0034                K = K + 1
0035          5     CONTINUE
0036                NCPR = K-1
0037                  IF(K.GT.NCMAT)RETURN
0038                DO 6 I = K,NCMAT
0039          6     CHL(I)=0
0040                RETURN
0041                END
```

ORIGINAL PAGE IS
OF POOR QUALITY

9.5 SUBROUTINE HOCUTT

```
HFORTRAN IV-PLUS V02-04                 13139149    31-AUG-77          PAGE 1
HOCUTT.FTN        /TRIGLOCKS/WR
0001                SUBROUTINE HOCUTT
0002                INCLUDE 'SYIFUL0ICOM.COM'
      *   C         THIS VERSION OF FUL0ICOM.COM IS TAYLORED TO FUL013,AND FUL014.
      *   C         THE CAMS HYBRID VERSIONS. TO CHANGE TO THE RTE VERSIONS, SET
      *   C         NCMAT TO 64 AND CHANGE THESE COMMENTS.
0003  *           IMPLICIT INTEGER(A - Z)
0004  *             PARAMETER NCMAT=8
0005  *           REAL GP,RFAC,XZ,YZ
0006  *           BYTE FILE,MGL,MGF,LGL,LGF,MFL,MFG,CHL,CHP,NC,MXC,LUT
0007  *           COMMON /FCOM/FILE(33),NC(5),MGL,MGF,LGL,LGF,MFL,MFG,CHL(NCMAT),
      *          1 CHP(NCMAT),MXC(NCMAT,5),NF,F11,MTXTFG,TUNIT,TUN,FLN,FORM,
      *          2 TTX1,TTY1,TTX2,TTY2,IIX1,IIY1,IIX2,IIY2,IX(512),IY(512),
      *            3 TX(512),TY(512),MX,MY,MX,NCMAX,LGBC(5,3),LUN,GP(NCMAT+1,5)
      *            4 ,LUT(256,5),NCP,RFAC,XZ,YZ
0008  *             COMMON/HCOM/SS,SF,LS,LE,NRDS,NDSPR,NCPR,NRPC,ANCL,
      *            1 NC,NS,NBIT,DBI,NCAF,SV,BSIZ,BSKIP,MSIZ,CALP,CERR
0009  *             COMMON /FATAL/ZZ,ZR
0010  *             COMMON /UCLP/IP,LP
0011  *           INTEGER FCOM(2702+27*NCMAT/2),HCOM(20)
0012  *           EQUIVALENCE (FCOM(1),FILE(1))
0013  *           EQUIVALENCE (HCOM(1),SS)
0014                REAL L,H,LL,HH
0015                LL = 32767
0016                HH = -32768
0017                DO 1 I = 1,5
0018                IF(LGBC(I,1).EQ.0)GO TO 1
0019                L = -LGBC(I,3)/100.
0020                H = L+FLOAT(256*NC)/FLOAT(LGBC(I,2))
0021                IF(I.EQ.(I/4)*4)H = L + 
0022                IF(I.EQ.(I/4)*4)L=H*2
0023                IF(L.LT.LL)LL=L
0024                IF(H.GT.HH)HH=H
0025          1     CONTINUE
0026                L = -LL
0027                H = FLOAT(256*NC)/(HH-LL)
0028                DO 2 I = 1,NC
0029                LGBC(I,2)=H+.5
0030                IF(I.EQ.(I/4)*4)LGBC(I,2)=2*LGBC(I,2)
0031                LGBC(I,3)=L*100.+.5
0032          2     IF(I.EQ.(I/4)*4)LGBC(I,3)=L*100.(I,3)/2
0033                RETURN
0034                END
```

9.6 PROGRAM FULOI2

```
      PROGRAM FUL0I2
        INCLUDE 'SY:FUL0ICOM.COM'
        BYTE BLOCK(0:9999),VB(512,5)
        DATA BUFBIG/10000/
        DATA LINLST/-1/
        DATA I0,LP/6,5/
        CALL DETACH
        CALL ATTACH
      OPEN(UNIT=1,NAME='[300,1]FUL0I.DAT',FORM='UNFORMATTED',
     1 TYPE='OLD',READONLY)
      READ(1)FCOM,MCOM
      CLOSE(UNIT=1)
        RFAC = RFAC * X2
D       WRITE(LP,1004)RFAC,X2
D1004   FORMAT(' RFAC, X2  ',2F10.5)
        BUFSZ = MSIZ*2
      IF(MSIZ*2.LT.RSIZ)BUFSZ=RSIZ
      BLOS2=BUFBIG-12-BUFSZ
      TB = TTX1-1
      IF(TTX1.GT.TTX2)TB=TTX2-1
      CALL DSET(1,F11,FILE,MTXTFG,TUNIT,0,1)
        TYG=(TY(1)/6)*6
D       WRITE(LP,1006)TYG
D1006   FORMAT(' TYG   ',I5)
      CALL MREAD(1,BLOCK,BLOCK(BLOSZ+1),BUFSZ,FORM,F11,PRTY,EOF,1)
      CALL FFIND(1,TTX1,TTY1,TTX2,TTY2,FORM,BLOCK(BLOSZ+1),BUFSZ)
      DO 1 L = 1,MY
      LINE=TY(L)
        IF(LINE.EQ.LINLST) GO TO 6
        LINLST = LIN
D       WRITE(LP,1000)LINE
1000    FORMAT(' LINE ',I10)
      DO 2 I = 1,NCR
        ICH = CHL(I)
        CALL LREED(1,BLOCK(NX*(I-1)+1),BLOCK(BLOSZ+1),RSIZ,LINE,F11,PRTY,
     1 EOF,FORM,ICH)
D       WRITE(LP,1007)I,ICH,(BLOCK(NX*(I-1)+J),J=1,NX)
D1007   FORMAT(' LREED  ',2I10/(' ',32(03,1X)))
2       IF(CERR.NE.0) GO TO 100
        LINE = ((LINE/6)*6-TYG)*RFAC+.5
D       WRITE(LP,1000)LINE
      DO 3 I = 1,5
D       WRITE(LP,1002)I,IY(L)
1002    FORMAT(' IRV ',2I10)
      IF(WC(I).NE.0)CALL IRV(I,IY(L),VB(1,I))
3       CONTINUE
        CALL WAIT
      DO 4 I = 1,MX
      IPIX=IX(I)-IP
        IXI=IX(I)-LINE*1
        IF(IXI.LE.0)GO TO 4
        IF(IXI.GT.512)GO TO 4
        CALL GBCALC(IPIX,IXI,BLOCK,VB,NX,NCR)
4       CONTINUE
6        DO 5 I =1,5
D       WRITE(LP,1003)I,IY(L)
1003    FORMAT(' IWV ',2I10)
      IF(WC(I).NE.0)CALL IWV(I,IY(L),VB(1,I))
5       CONTINUE
        CALL WAIT
1     CONTINUE
100     CONTINUE
        CALL DUSET(1,F11,1)
        CALL SETEF(53)
```

```
  CALL DETACH
CALL EXIT
END
```

9.7 SUBROUTINE GBCALC

```
      SUBROUTINE GBCALC(TPIX,IPIX,BUF,VL,NXX,NCR)
        INCLUDE 'SY:FUL0IC0M.C0M'
        REAL Q
      REAL AD(5)
      BYTE VL(512,5)
      BYTE BUF(NXX,NCR)
      IF(MGF.EQ.0) G0 T0 1
      IF(LGF.NE.0)RETURN
      D0 2 I = 1,5
      C = LGBC(I,1)
      IF(C.EQ.0)G0 T0 2
        C=CHP(C)
        J=FFUNC(BUF(TPIX,C))
        VL(IPI ,I)=LUT(J+1,I)
2     CONTINUE
      RETURN
1     D0 3 IC = 1,5
      Q = 0
      D0 4 I = 1,NCMAT
      C = MXC(I,IC)
      IF(C.EQ.0) G0 T0 5
        C=CHP(C)
4     Q = Q+FFUNC(BUF(TPIX,C))*GB(I,IC)
5     IF(I.EQ.1) G0 T0 6
      AD(IC)=Q+GB(NCMAT+1,IC)
      G0 T0 3
6     AD(IC)=0
3     CONTINUE
      D0 8 I = 1,5
      IF(LGF.NE.0) G0 T0 7
      C = LGBC(I,1)
      IF(C.EQ.0) G0 T0 8
        AD(I)=FL0AT(LGBC(I,2))/100.*(AD(C)+FL0AT(LGBC(I,3))/100.)
7       IF(AD(I).GT.255.)AD(I)=255.
        IF(AD(I).LT.0.)AD(I)=0.
        VL(IPIX,I)=AD(I)
8     CONTINUE
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

9. IMAGE DISPLAY

This program has been extracted from the more comprehensive Consolidate Tape Read Program which is described in reference 1. Flow charts are not available for subroutines CRUNCH (section 9.1), GABI (section 9.2), COMLUT (section 9.3), TCHLST (section 9.4), HOCUTT (section 9.5), FULOI2 (section 9.6), and GBCALC (section 9.7).

10. FIELD DEFINITION

This program is an adaptation of the more comprehensive Irregular Cursor Program, IRREG, which is described in reference 2.

10. FIELD DEFINITION FLDDEF

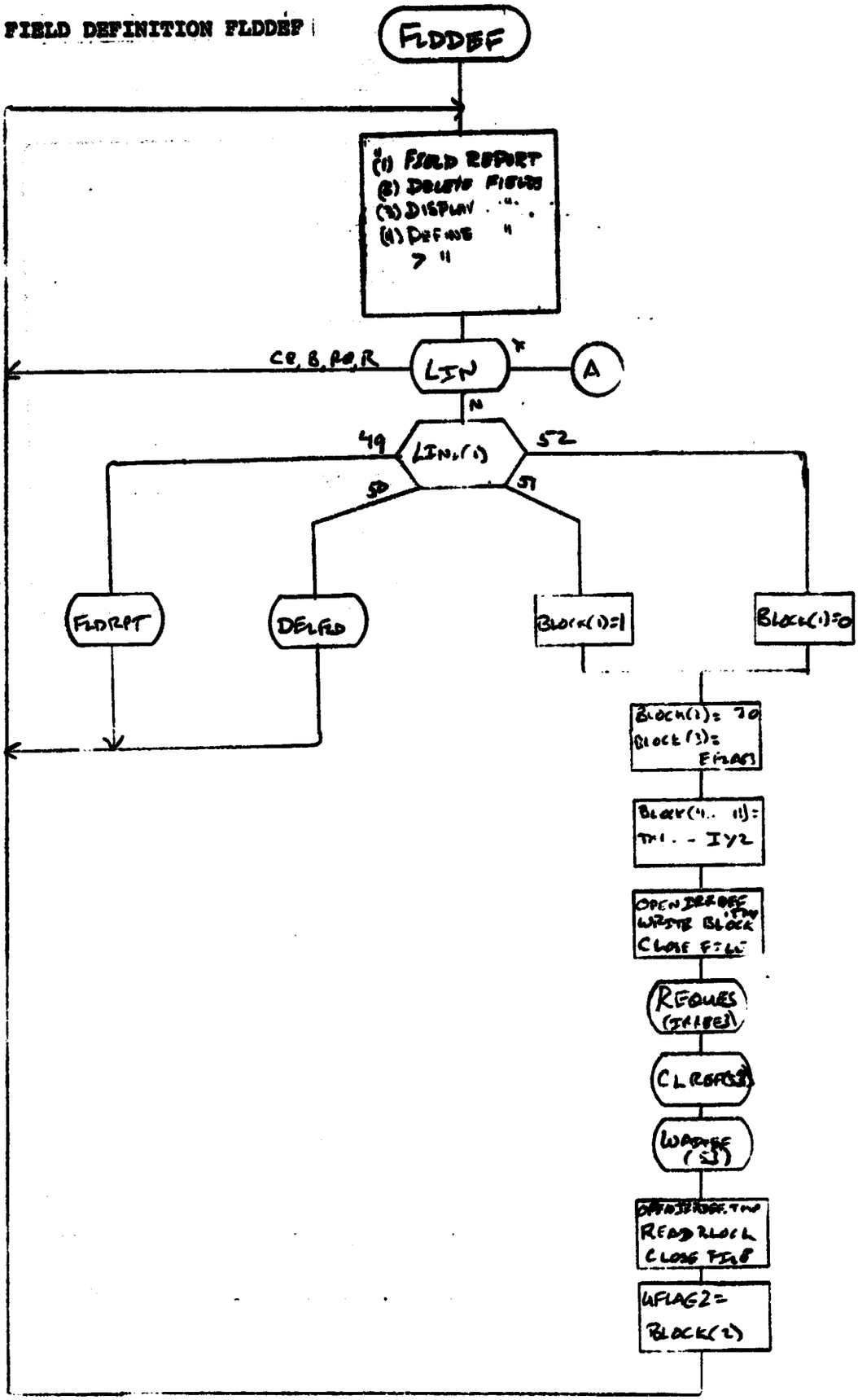
```
HP2RTRAN IV+PLUS V02-04              12:32:59    29-AUG-77          PAGE 1
FLDDEF.FTN       /THISBLOCKS/WP
0001              PROGRAM FLDDEF
0002              IMPLICIT INTEGER(A-Z)
0003              INCLUDE '[300,3]CANSC0M2.INC'
0004  *           INCLUDE 'SYS[300,3]CANSPARAM.INC'
0005  *           PARAMETER MAXCAT=80,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      *          1,MAXV=11,NDATS=200,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *          2N0SPWD=6,N0DTW0=10
0006  *           EQUIVALENCE (C1,ACDATE),(C2,ISEG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0007  *           INTEGER C1(466),C2(256),C3(71),C4(348),C5(629)
      *     C*
0008  *           INTEGER ACDATE,SUBCAT,SUBP2P,CATKNT,CATTH
0009  *           BYTE CHNVEC,N0CHAN,N0SUB,D0TCAT,D0TCLU
0010  *           C0MM0N/C0M1/ACDATE(2,MAXACC),CHNVEC(MAXCHN,MAXACC),N0CHAN,N0SUB,
      *          1SUBCAT(MAXSUB),SUBP2P(MAXSUB),CATKNT(MAXCAT),CATTH(MAXCAT),N0D0,
      *          2N0DU,N0TH,D0TCAT(NDATS),D0TCLU(NDATS)
      *     C*
0011  *           INTEGER ADATES,SUBAZ,ANALST,FLDDAY,D2TDAY,PDATE1,TDATE1
0012  *           INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RAND0M,GRID
0013  *           BYTE DELFLG,N0ACD,SAILGR,SAMEL,NSTART,NTYPE1,ALP,ALP0
0014  *           BYTE PCTCT,PCTCT2,VAR,VAR2,DLABEL,TYPE
0015  *           C0MM0N/C0M2/ISEG,DELFLG,N0ACD,ADATES(2,MAXACD),SAILGR(MAXACD),
      *          1SUMEL(MAXACD),SUMAZ(MAXACD),IMDATE(2),ANALST(5),FLDDAY(2),
      *          2D2TDAY(2),NSTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      *          3PDATE3(2),TDATE3(2),MAXCAT,CATNAM(MAXCAT),ALP(MAXCAT),ALP0,
      *          4          PCTCT(MAXCAT),PCTCT2,VAR(MAXCAT),VAR2
      *     C*
0016  *           INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      *          1UFLAG4
0017  *           INTEGER PFLAG,TSKMNT
0018  *           C0MM0N/C0M3/PFLAG,TSKMNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
      *          1,UFLAG2,UFLAG3,UFLAG4,NELMB(MAXSUB)
      *     C*
0019  *           INTEGER TX1,TY1,TX2,TY2,ACTISP,G,R,DTAIND,D2TARY,GMIN,GMAX,FUL
0020  *           INTEGER SPWIND,CLAWND,CLUWND
0021  *           C0MM0N/C0M4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACDISP(2),II1(4),G(4),
      *          1R(4),DTAIND(5,N0DTWD),SPWIND(5,N0SPWD),IMWIND(4),NUMD0T,
      *          2D2TARY(D0TS),GMIN,GMAX,FUL(2,7),CLAWND(8),CLUWND(8)
0022  *           C0MM0N/C0M5/DISKID,RANDOM(NDATS),GRID(NDATS),DLABEL(NDATS),
      *          1TYPE(NDATS),PRCL20
                  BYTE LINE(74)
                  INTEGER BLALBL(NDATS)
                  INTEGER BLOCK(15),IRPEG3(2),DLBL(MAXFLD),
                 1 IV(MAXFLD)
                  BYTE FIELD(2*MAXV,MAXFLD),I1,I2
                  INTEGER RATS(2*MAXV,MAXFLD)
                  DATA I0/6/
                  DATA IRPEG3/3RIMG,3REG3/
                  II = I0
                  IPA = 1
0032      1       CALL OSGRPH(I2,IPA,'FIELD DEFINITION PROGRAM 1.0 5/77',33)
0033              WRITE(I2,1000)
0034      1000    FORMAT(' (1) FIELD REPORT'/' (2) DELETE FIELDS'/
                  1 ' (3) DISPLAY FIELDS'/' (4) DEFINE NEW FIELDS'/
                  2 ' (X) EXIT'/'$> ')
0035              CALL LIN(I0,LINE,TI,IF)
0036              GO TO (99,1,1,1,1,2),IF
```

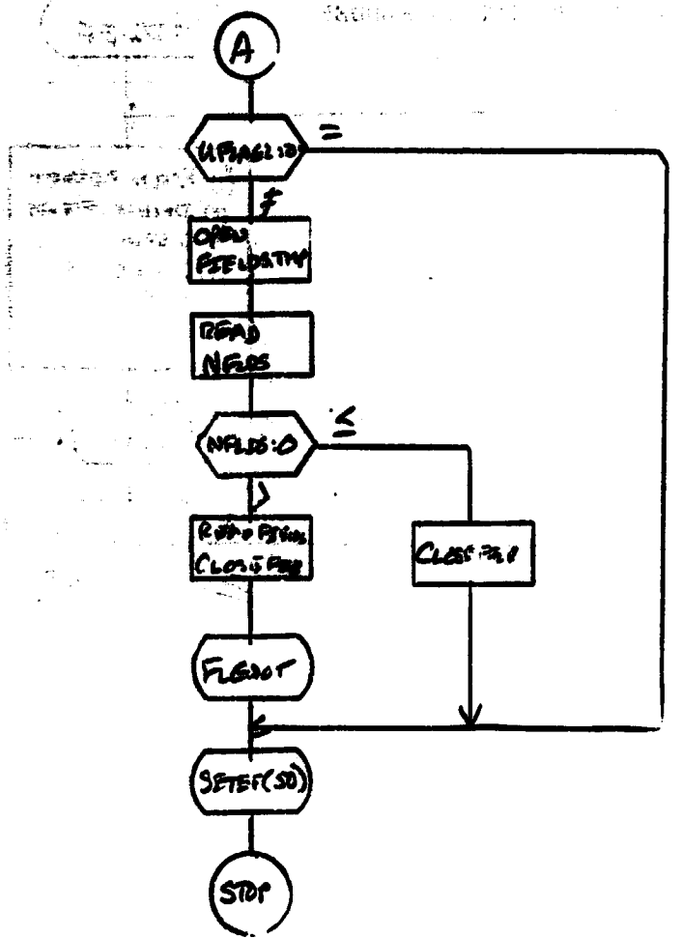

BEST AVAILABLE COPY
QUALITY

```
FLDDEF.FTN      /TR:BLØCKS/WR
0037    2       GØ TØ (3,4,5,5),LINE(1)-*60
0038            GØ TØ 1
0039    3       CALL FLDRPT(15)
0040            GØ TØ 1
0041    4       CALL DELFLD(10)
0042            GØ TØ 1
0043    5       PLØCK(1)=52-LINE(1)
0044            BLØCK(2)=I0
0045            BLØCK(3)=EFLAG3
0046            BLØCK(4)=TX1
0047            BLØCK(5)=TY1
0048            BLØCK(6)=TX2
0049            BLØCK(7)=TY2
0050            BLØCK(8)=IX1
0051            BLØCK(9)=IY1
0052            BLØCK(10)=IX2
0053            BLØCK(11)=IY2
0054            ØPEN (UNIT=1,NAME='[300,1]IRØDEF.TMP',TYPE='UNKNØWN',
                1 FØRM='UNFØRMATTED')
0055            WRITE(1)BLØCK
0056            CLØSE(UNIT=1)
0057            CALL RØQUES(IRPEG3)
0058            CALL CLREF(53)
0059            CALL WAITFR(53)
0060            ØPEN (UNIT=1,NAME='[300,1]IRØDEF.TMP',TYPE='UNKNØWN',
                1 FØRM='UNFØRMATTED')
0061            READ(1)BLØCK
0062            CLØSE(UNIT=1)
0063            IF(EFLAG3.EQ.0)EFLAG3=BLØCK(2)
0064            UFLAG2=BLØCK(2)
0065            GØ T  1
0066    99      IF(UFLAG2.EQ.0)GØ TØ 100
0067            ØPEN(UNIT=1,NAME='[300,1]FIELDS.TMP',TYPE='ØLD',
                1 FØRM='UNFØRMATTED')
0068            READ(1)NFLD
0069            IF(NFLD.EQ.0) GØ TØ 1234
0070            DØ 101 I=1,NFLD
0071            READ(1)(PLØCK(J),J=1,3),I1,I2,(FIELD(J,I),J=1,
                1 2*MAXV)
0072            NV(I)=I2
0073            DØDØ(I)=I1
0074    101     CØNTINUE
0075            DØ 102 I = 1,NFLD
0076            DØ 102 J = 1,2*MAXV
0077    102     DATS(J,I)=FFUNC(FIELD(J,I))
0078            CLØSE(UNIT=1)
0079            DØ 6543 I = 1,NØDTS
0080    6543    BLABEL(I)=ØLABEL(I)
0081            CALL FLGRØT(NFLD,NV,DATS,DØDØ,BLABEL)
0082            UFLAG3=1
0083            DØ 6542 I = 1,NØDTS
0084    6542    OLABEL(I)=BLABEL(I)
0085            GØ TØ 100
0086    1234    CLØSE(UNIT=1)
0087    100     CØNTINUE
0088            INCLUDE '[300,3]CAMSAVE.INC'
```

```
FORTRAN IV-PLUS V02-04          12132159    29-AUG-77          PAGE 3
FLDDEF,FTN      /TR:SL@CK$/WR
0089 *          @PEN(UNIT=1,NAME='[300,1]GLOBAL.TMP;1',FORM='UNFORMATTED',
     *        1   TYPE='UNKNOWN',ERR=9999)
0090 *            WRITE(1)C1
0091 *            WRITE(1)C2
0092 *            WRITE(1)C3
0093 *            WRITE(1)C4
0094 *            WRITE(1)C5
0095 *            CLOSE(UNIT=1)
0096 *            GO TO 9991
0097 *  9999      TYPE 9990
0098 *  9990      FORMAT(1X,'OPEN FAILURE ON [300,1]GLOBAL.TMP--NO RESTART')
0099 *  9991      CONTINUE
0100            CALL SETEF(50)
0101            END
```

10. FIELD DEFINITION FLDDEF




A
=
READ
NFLDS
≤
FLGNOT
STOP

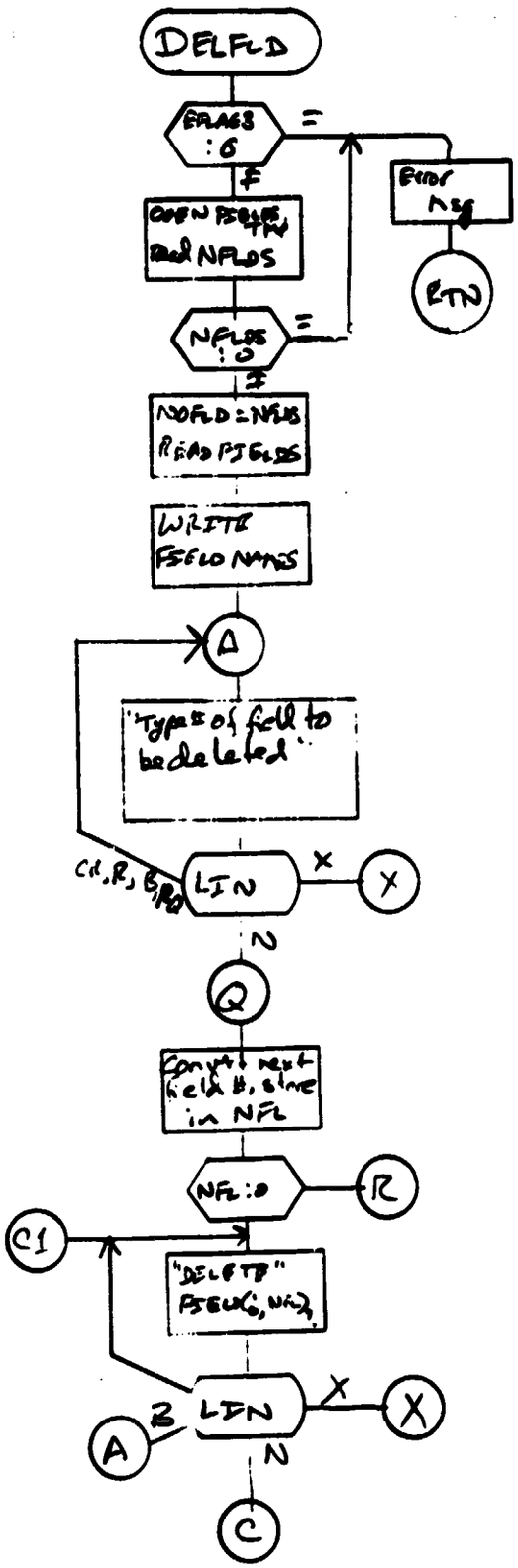
10.1 SUBROUTINE DELFLD

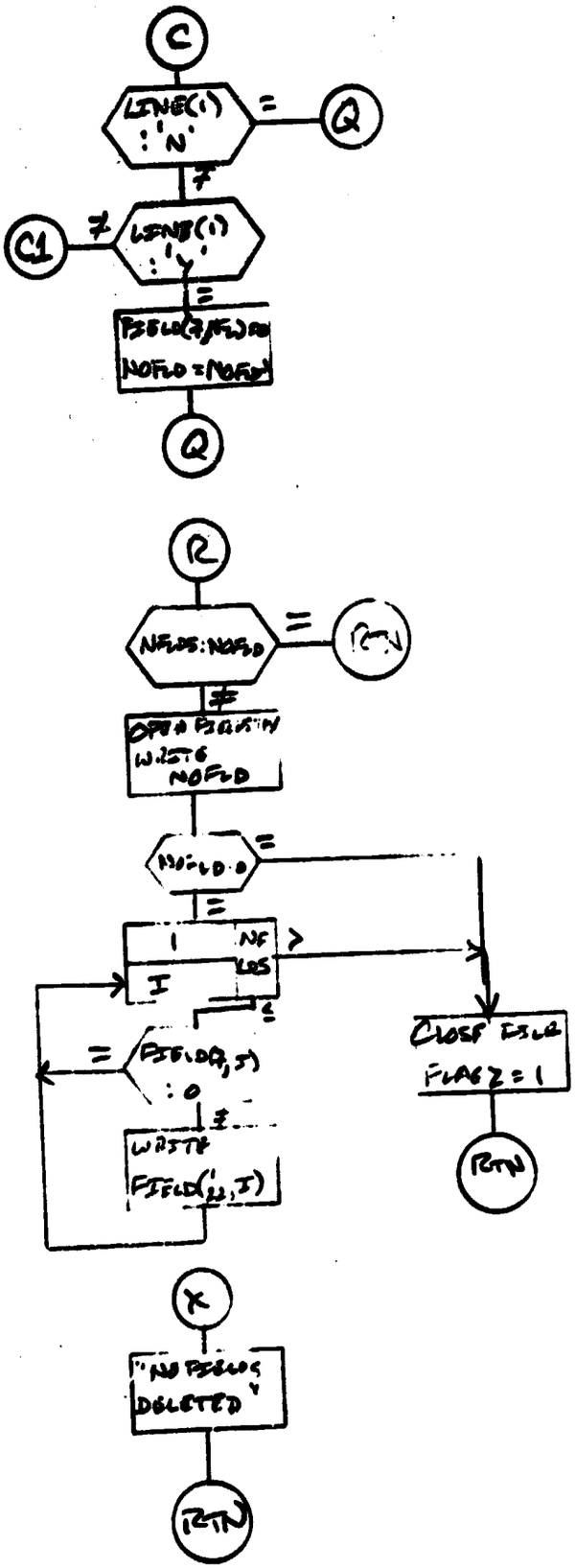
```
MFORTRAN IV PLUS V02-04          12:33:29   29-AUG-77          PAGE 1
DELFLD.FTN      /TR:BLOCKS/WR
0001              SUBROUTINE DELFLD(I0)
0002              IMPLICIT INTEGER (A-Z)
0003              INCLUDE '[300,3]CAMSCOMON.INC'
0004  *         INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0005  *         PARAMETER MAXCAT=60,MAXSUB=62,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=90
      *        1,MAXV=11,NDOTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *        2NOSPWD=6,NODTWD=10
0006  *         EQUIVALENCE (C1,ACDATE),(C2,ISEG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0007  *         INTEGER C1(469),C2(256),C3(71),C4(348),C5(629)
      *   C*
0008  *         INTEGER ACDATE,SUBCAT,SUBPOP,CATNAT,CATTH
0009  *         BYTE CHNVEC,NOCHAN,NOSUB,DOTCAT,DOTCLU
0010  *         COMMON/COM1/ACDATE(2,MAXACC),CHNVEC(MAXCHN,MAXACC),NOCHAN,NOSUB,
      *        1SUBCAT(MAXSUB),SUBPOP(MAXSUB),CATNAT(MAXCAT),CATTH(MAXCAT),NODO,
      *        2NODU,NPTH,DOTCAT(NDOTS),DOTCLU(NDOTS)
      *   C*
0011  *         INTEGER ADATES,SUMAZ,ANALST,FLDDAY,DOTDAY,PDATE1,TDATE1
0012  *         INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RANDOM,GRID
0013  *         BYTE DELFLG,NOACQ,SOILGR,SUNEL,ASTART,NTYPE1,ALP,ALP0
0014  *         BYTE PCTCT,PCTCT0,VAR,VAR0,DLABEL,TYPE
0015  *         COMMON/COM2/ISEG,DELFLG,NOACQ,ADATES(2,MAXACD),SOILGR(MAXACD),
      *        1SUNEL(MAXACD),SUMAZ(MAXACD),IMDATE(2),ANALST(5),FLDDAY(2),
      *        2DOTDAY(2),NSTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      *        3PDATE3(2),TDATE3(2),NOCAT,CATNAM(MAXCAT),ALP(MAXCAT),ALP0,
      *        4          PCTCT(MAXCAT),PCTCT0,VAR(MAXCAT),VAR0
      *   C*
0016  *         INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      *        1UFLAG4
0017  *         INTEGER PFLAG,DSKMNT
0018  *         COMMON/COM3/PFLAG,DSKMNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
      *        1,UFLAG2,UFLAG3,UFLAG4,NEWLAB(MAXSUB)
      *   C*
0019  *         INTEGER TX1,TY1,TX2,TY2,ACDISP,G,E,DTWIND,DOTARY,GMIN,GMAX,FUL
0020  *         INTEGER SPWIND,CLAWND,CLUWND
0021  *         COMMON/COM4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACDISP(2),I1(4),G(4),
      *        1E(4),DTWIND(5,NODTWD),SPWIND(5,NOSPWD),IMWIND(4),NUMDOT,
      *        2DOTARY(NDOTS),GMIN,GMAX,FUL(2,9),CLAWND(8),CLUWND(8)
0022  *         COMMON/COM5/DISKID,RANDOM(NDOTS),GRID(NDOTS),DLABEL(NDOTS),
      *        1TYPE(NDOTS),RECLOC
0023              BYTE FIELD(10*2*(MAXV-1),50),LINE(74)
0024              BYTE IA
0025              IF(PFLAG3.EQ.0)GO TO 999
0026              OPEN(UNIT=1,NAME='[300,1]FIELDS.TMP',TYPE='OLD',FORM=
                  1 'UNFORMATTED')
0027              READ(1)NFLDS
0028              IF(NFLDS.EQ.0) GO TO 10
0029              NOFLD=NFLDS
0030              DO 1 I = 1,NFLDS
0031    1         READ(1)(FIELD(J,I),J=1,10*2*(MAXV-1))
0032              CLOSE(UNIT=1)
0033    3         WRITE(10,1000)(I,(FIELD(J,I),J=1,6),I=1,NFLDS)
0034    1000      FORMAT(' CURRENT FIELDS ARE:'/(' ',5('(',I2,') ',6A1,
                  1 2X)))
0035              T=10
0036    2         WRITE(10,1001)
0037    1001      FORMAT('$TYPE NUMBERS OF FIELDS TO BE DELETED  > ')
```

```
DELFLD.FTN      /TR:BLOCKS/WD
0038            CALL LIN(10,LINE,71,IF)
0039            IP=0
0040            GO TO (99,3,2,2,2,4),IF
0041    4       CALL INTFF(IP,LINE,74,1)
0042    D       WRITE(6,1005)I,IP,(LINE(IG),IG=1,40)
        D1005   FORMAT(' ',2I5,'!',40A1,'!!!')
0042            IF(I.GT.NFLDS)GO TO 4
0043            IF(I.LE.0) GO TO 5
0044    6       WRITE(10,1002)(FIELD(J,I),J=1,6)
0045    1002    FORMAT('$DELETE ',6A1,' (Y)ES/(N)O  > ')
0046            READ(10,1006)IA
0047    1006    FORMAT(A1)
0048            IF(IA.EQ.'X')GOTO99
0049            IF(IA.EQ.'B')GOTO2
0050            IF(IA.EQ.'N')GOTO4
0051            IF(IA.NE.'Y')GOTO6
0052            FIELD(7,I)=0
0053            NOFLD=NOFLD+1
0054            GO TO 4
0055    5       IF(NFLDS.EQ.NOFLD)RETURN
0056            OPEN(UNIT=1,NAME='[300,11FIELDS.TMP',TYPE='OLD',FORM=
                1 'UNFORMATTED')
0057            WRITE(1)NOFLD
0058            IF(NOFLD.EQ.0) GO TO 9
0059            DO 8 I = 1,NFLDS
0060    8       IF(FIELD(7,I).NE.0)WRITE(1)(FIELD(J,I),J=1,2*(MAXV+1)+10)
0061    9       CLOSE (UNIT=1)
0062            UFLAG2=1
0063            RETURN
0064    999     WRITE(10,1003)
0065    1003    FORMAT(' EXISTANCE FLAG FOR FILE "FIELDS.TMP" NOT SET'/
                1 ' CLEARLY NO FIELDS CAN BE DELETED.....'///)
0066            RETURN
0067    99      WRITE(10,1004)
0068    1004    FORMAT(' USER TYPED "X", GOODBYE, NO FIELDS DELETED.'///)
0069            RETURN
0070    10      CLOSE(UNIT=1)
0071            WRITE(10,1005)
0072    1005    FORMAT(' NO FIELDS TO DELETE.'////)
0073            RETURN
0074            END
```

ORIGINAL PAGE IS OF POOR QUALITY

10.1 SUBROUTINE DELFLD




C
LINE(1) : 'N'
=
Q
≠
C1
≠
LINE(1) : 'Y'
=
NOFLD = NOFLD
Q
R
NFLDS : NOFLD
=
RTN
≠
WRITE
NOFLD
NOFLD : 0
=
1
NF
LOS
>
I
=
FIELD(I) : 0
=
≠
WRITE
FIELD(I)
CLOSE FILE
FLAG2 = 1
RTN
X
'NO FIELDS DELETED'
RTN

10.2 SUBROUTINE FLDRPT

```
HP3000 FTN IV-PLUS V02-04          12:34:28  29-AUG-77          PAGE 1
FLDRPT.FTN       /TRIBLOCKS/WR
0001               SUBROUTINE FLDRPT
0002               IMPLICIT INTEGER(A-Z)
0003             INCLUDE '[300,3]CAMSCOMON.INC'
0004  *          INCLUDE '$YS[300,3]CAMSPARAM.INC'
0005  *          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=90
      *         1,MAXV=11,NDBTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *         2NBSPWD=6,NBDTWD=10
0006  *          EQUIVALENCE (C1,ACDATE),(C2,ISEG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0007  *          INTEGER C1(469),C2(256),C3(71),C4(348),C5(629)
      *   C*
0008  *          INTEGER ACDATE,SUBCAT,SUBPOP,CATKNT,CATTH
0009  *          BYTE CHNVEC,NOCHAN,NOSUB,DBTCAT,DBTCLU
0010  *          COMMON/C0M1/ACDATE(2,MAXACC),CHNVEC(MAXCHN,MAXACC),NOCHAN,NOSUB,
      *         1SUBCAT(MAXSUB),SUBPOP(MAXSUB),CATKNT(MAXCAT),CATTH(MAXCAT),NODB,
      *         2NODU,NDTH,DBTCAT(NDBTS),DBTCLU(NDBTS)
      *   C*
0011  *          INTEGER ADATES,SUNAZ,ANALST,FLDDAY,DBTDAY,PDATE1,TDATE1
0012  *          INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RANDOM,GRID
0013  *          BYTE DELFLG,NOACQ,SOILOR,SUNEL,NSTART,NTYPE1,ALP,ALPO
0014  *          BYTE PCTCT,PCTCTO,VAR,VARO,DLABEL,TYPE
0015  *          COMMON/C0M2/ISEG,DELFLG,NOACQ,ADATES(2,MAXACD),SOILOR(MAXACD),
      *         1SUNEL(MAXACD),SUNAZ(MAXACD),IMDATE(2),ANALST(5),FLDDAY(2),
      *         2DBTDAY(2),NSTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      *         3PDATE3(2),TDATE3(2),NOCAT,CATNAM(MAXCAT),ALP(MAXCAT),ALPO,
      *         4          PCTCT(MAXCAT),PCTCTO,VAR(MAXCAT),VARO
      *   C*
0016  *          INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      *         1UFLAG4
0017  *          INTEGER PFLAG,DSKMNT
0018  *          COMMON/C0M3/PFLAG,DSKMNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
      *         1,UFLAG2,UFLAG3,UFLAG4,NEWLAB(MAXSUB)
      *   C*
0019  *          INTEGER TX1,TY1,TX2,TY2,ACDISP,G,B,DTWIND,DBTARY,GMIN,GMAX,FUL
0020  *          INTEGER SPWIND,CLAWND,CLUWND
0021  *          COMMON/C0M4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACDISP(2),I1(4),G(4),
      *         1B(4),DTWIND(5,NODTWD),SPWIND(5,NOSPWD),IMWIND(4),NUMDBT,
      *         2DBTARY(NDBTS),GMIN,GMAX,FUL(2,7),CLAWND(8),CLUWND(8)
0022  *          COMMON/C0M5/DISKID,RANDOM(NDBTS),GRID(NDBTS),DLABEL(NDBTS),
      *         1TYPE(NDBTS),RECLPC
0023               BYTE BVTX,DEV,NV,LABEL
0024               DIMENSION FLDNAM(3),VERTEX(2,MAXV),TYP(3)
0025               DIMENSION BVTX(2,MAXV)
0026             DATA DU/2HDU/
0027             DATA DO/2HDO/
0028             DATA TYP/2H  ,2H  ,2H  /
0029               ID=6
0030     1       CALL OUTPUT(7)
0031               CALL OUTPUT(27,12)
0032               WRITE(ID,10)
0033     10      FORMAT('$ REPORT DEVICE (T)ERMINAL,(G)RULD OR (L)INE PRINTER?>')
0034             READ(ID,20)DEV
0035     20      FORMAT(A1)
0036             IF(DEV.EQ.'B'.OR.DEV.EQ.'X') RETURN
0037             IF(DEV.EQ.'T') GO TO 100
0038             IF(DEV.EQ.'G') GO TO 101
0039             IF(DEV.NE.'L') GO TO 1
```

```
FEDRPT.FTN      /TR:BL:CKS/NR
0040              I0=8
0041              CALL ASSIGN(I0,'LP:')
0042              MLINE=18
0043              WRITE(I0,1008)
0044              GO TO 102
0045        101   I0=9
0046              CALL ASSIGN(I0,'CP:')
0047              WRITE(I0,1008)
0048              MLINE=18
0049            GO TO 102
0050        100   I0=10
0051              CALL ASSIGN(I0,'TI:')
0052              MLINE=5
0053        102   IP=1
0054              IF(EFLAG3,EQ,0) GO TO 300
0055              OPEN(UNIT=7,NAME='[300,10]FIELDS.TMP',ACCESS='SEQUENTIAL',
                  1 TYPE='UNKNOWN',FORM='UNFORMATTED')
0056              READ(7,END=400,ERR=400) NOFLD
0057              IF(NOFLD,EQ,0) GO TO 400
0058            CALL CSGOPN(I0,IP,'FIELD DEFINITION REPORT ',24)
0059            WRITE(I0,1000)
0060        1000  FORMAT(' FIELD/          1ST   2ND   3RD   4TH   5TH   6TH   7TH   8
                XTH   9TH  10TH ')
0061            WRITE(I0,1001)
0062        1001  FORMAT('  TYPE          ',10(' VERTX'))
0063            WRITE(I0,1002)
0064        1002   FORMAT('  ')
0065            ICNT=1
0066            DO 200 I=1,NOFLD
0067            IF(ICNT,GT,MLINE) GO TO 900
0068        500   READ(7) FLDNAM,LABEL,NV,((VERTX(J,K),J=1,2),K=1,MAXV)
0069            IF(LABEL,EQ,-1) TYP(2)=D2
0070            IF(LABEL,EQ,-2) TYP(2)=DU
0071              DO 199 J=1,MAXV
0072              DO 198 K=1,2
0073              VERTEX(K,J)=IRVTE(0,VVTX(K,J))
0074        198     CONTINUE
0075        199     CONTINUE
0076            WRITE(I0,1003) FLDNAM,(VERTEX(2,K),K=1,NV-1)
0077        1003  FORMAT(' ',3A2,'  LINE',10I6)
0078            WRITE(I0,1004) TYP,(VERTEX(1,K),K=1,NV-1)
0079        1004  FORMAT(' ',3A2,'  PIXL',10I6)
0080            WRITE(I0,1002)
0081            ICNT=ICNT+1
0082        200   CONTINUE
0083              WRITE(ID,1009)
0084        1009    FORMAT('  END OF REPORT ,')
0085        201     CONTINUE
0086              CLOSE(UNIT=I7)
0087              CLOSE(UNIT=7)
0088        300     CALL OUTPUT(7)
0089              WRITE(ID,1007)
0090        1007    FORMAT('$ (R)ESTART OR E(X)IT?>')
0091              READ(ID,1006)DEV
0092              IF(DEV,EQ,'R') GO TO 1
0093              RETURN
```


ORIGINAL PAGE IS
OF POOR QUALITY

FEDRPT.FTN /TR:BLOCKS/WP

```
0094     900      IF(I0.GT.9)G0 T0 902
0095              WRITE(I0,1008)
0096     1008     F0RMAT(1H1)
0097              G0 T0 901
0098     902      C0NTINUE
0099              CALL 0UTPUT(7)
0100              WRITE(ID,1005)
0101     1005     F0RMAT('SENTER "F" F0R PAGE F0RWARD,"R" F0R RESTART 0R ',
                1'"X" F0R EXIT>')
0102              READ(ID,1006)DEV
0103     1006     F0RMAT(A1)
0104              IF(DEV.NE.'F') G0 T0 201
0105     901      CALL 0UTPUT(27,12)
0106              CALL CSGRPH(I0,IP,'FIELD DEFINITI0N REP0RT ',24)
0107              WRITE(I0,1000)
0108              WRITE(I0,1001)
0109             ICNT=0
0110             WRITE(I2,1002)
0111             G0 T0 500
0112     400      WRITE(ID,1010)
0113     1010     F0RMAT(' FIELD DEFINITI0N FILE EMPTY ')
0114              G0 T0 201
0115             END
```

10.3 SUBROUTINE IRREG3

```
MFØRTRAN IV-PLUS V02-04          12:35:01   29-AUG-77          PAGE 1
IRREG3.FTN       /TR:BLØCKS/WR
      C      PRØGRAM PØLYCR
0001             INCLUDE '[300,3]CAMSPARAM.INC'
0002  *          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      *         1,MAXV=11,NDØTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *         2NØSPWD=6,N2DTWD=10
0003             CØMMØN /CURCØM/X,Y,N,THM,M,B,IGMB,LINS(30,30),AA
                   1 ,IDM(3287)
0004             CØMMØN/FAKCØM/RI(2),RE(2),RSF(2)
                 * ,IP,CURFIL
0005             BYTE ASFILE(30)
0006               INTEGER ØS,FPTR,DDIESF
0007             REAL M(30),B(30)
0008               REAL MM
0009             INTEGER I,J,N,DØNE,CURFIL,L1,L2,MIN,X(30),Y(30)
                   1 ,IGMB(30),AA(37),P
0010               BYTE FIELD(10+2*(MAXV-1),MAXFLD)
0011             INTEGER THM,XJMXI,SD
0012               INTEGER BLØCK(15),SENDØR(2)
0013               DATA SENDØR/3RFLD,3RDEF/
0014             CALL ATTACH
0015               ØPEN(UNIT=1,NAME='[300,1]IRRDEF.TMP',TYPE='ØLD',FØRM=
                   1 'UNFØRMATTED')
0016               READ(1)BLØCK
0017               CLØSE(UNIT=1)
0018               ØS = '0'
0019               IPA = 1
0020               IPAINT = 0
0021               IP = 6
0022               THM = 1
0023               I0 = IP
0024     1         CALL CSGØPH(IP,IPA,'FIELD DISPLAY/DEFINE 1,0 5/77',29)
0025               IFST = 0
0026     41        CALL FAKCU3(BLØCK)
0027     401       NFL = 0
0028     42        CALL CSGØPH(IP,IPA,'FIELD DISPLAY/DEFINE 1,0 5/77',29)
0029               CALL FLDNAM(NFLDS,FPTR,NFL,CURFIL,DDIESF,IFST,FIELD,IP,NFLDS,BLØCK)
0030               IF(NFLDS.EQ.0)GØ TØ 505
0031               IF(CURFIL.EQ.2.AND.IFST.NE.1) GØ TØ 777
0032     38        WRITE(6,39)ØS
0033        39 FØRMAT(/'$(Ø)UTLINE FIELD ØR (S)HADE AREA  ',A1,'  > ')
0034               I0 = IP
0035               CALL LIN(IP,AA,I0,IF)
0036               GØ TØ (505,401,1,38,45,402),IF
0037     402       CØNTINUE
0038               ØS=AA(1)
0039             IF(AA(1).EQ.'0')GØ TØ 46
0040             IF(AA(1).NE.'S')GØ TØ 38
0041             IPAINT=1
0042             GØ TØ 45
0043        46 IPAINT=0
0044        45 CØNTINUE
0045     47        IF(CURFIL.EQ.0)CALL SWCLP
0046     48        WRITE(6,1000)THM
0047     1000      FØRMAT('$DISPLAY ØN THEME NUMBER ?  ',I1,'  > ')
      C*     READ THEME NUMBER
0048               I0 = IP
```

```
0049            CALL LIN(IP,AA,I?,IF)
0050            GØ TØ (505,38,1,43,777,40),IF
0051    40      P=0
0052            CALL INTFF(P,AA,74,THM)
0053            IF(THM.LE.0.ØR.THM.GT.8) GØ TØ 47
0054    777     I = 0
0055          N = 0
0056          DØNE = 0
0057    3     CALL RØXYD3(IX,IY,DØNE,IDEL,CURFIL,FPTR,FIELD,N)
0058          IF(DØNE.NE.0) GØ T? 6
0059          IF(IDEL.NE.1) GØ TØ 4
0060    22    SD =0
0061            NN = N
0062            IF(DØNE.EQ.1) N = N + 1
0063            IF(N.LE.0) GØ TØ 3
0064            IF(N.EQ.1) GØ TØ 27
0065          CALL SDLINE(SD)
0066    27      CALL SDPNT( X(1),Y(1),N,SD,THM)
0067          N = N-1
0068            DØNE = 0
0069          I = I - 1
0070          GØ T? 33
0071    20      WRITE(6,1007)(MAXV-1)
0072    1007    FØRMAT(' YØU HAVE ATTEMPTED TØ ENTER MØRE THAN ',I2,' CØRNERS,'/
                1 ' THE LAST CORNER WAS IGNØRED.'/' YØU ARE DØNE.'//)
0073            CALL OUTPUT(7,7,7)
0074            N = N - 1
0075            DØNE = 1
0076            GØ T? 6
0077    4       IF(N.LE.0) GØ TØ 44
0078            IF(X(N).EQ.IX.AND.Y(N).EQ.IY) GØ T? 3
0079    44      N = N + 1
0080            IF(N.GT.(MAXV-1)) GØ T? 20
0081          X(N)=IX
0082          Y(N)=IY
0083            SD = 1
0084            CALL SDPNT(X,Y,N,SD,THM)
0085    6     IF(N.GE.2) GØ TØ 7
0086            IF(N.EQ.1.AND.DØNE.EQ.1)GØ T? 7
0087          DØNE = 0
0088    33      CALL INS("200)
0089          GØ T? 3
0090    7       NN = N
0091            SD = 1
0092          IF(DØNE.EQ.0) GØ T? 8
0093          I = N
0094          J = 1
0095            N = N + 1
0096            X(N) = X(1)
0097            Y(N) = Y(1)
0098          GØ TØ 9
0099    8     I = N-1
0100          J = 1
0101    9     XJMXI=X(J)-X(I)
0102          IF(XJMXI.EQ.0) GØ T? 10
0103          IGMB(I)=0
```

```
0104            M(I)=FLØAT(Y(J)-Y(I))/XJMXI
0105            B(I)=Y(I)-M(I)*X(I)
0106            GØ TØ 11
0107      10    IGMB(I)=1
0108            M(I)=0
0109            B(I)=0
0110      11    CALL SPLINE(SD)
0111              N = NN
0112            IF(N.GT.3) GØ TØ 12
0113            IF(N.EQ.3.AND.DØNE.EQ.1) GØ TØ 12
0114              IF(N.LT.3.AND.DØNE.EQ.1)GØ TØ 500
0115            GØ TØ 33
0116      12    L1 = I
0117            L2 = I-2
0118            MIN = 1+DØNE
0119      24    IF(L2.GE.MIN) GØ TØ 13
0120            IF(DØNE.NE.1) GØ TØ 3
0121            GØ TØ 500
0122      13    L1P=L1+1
0123            L2P=L2+1
0124            IF(L1P.GT.N)L1P=1
0125            IF(L2P.GT.N)L2P=1
0126            IF(IGMB(L1).EQ.0) GØ TØ 14
0127            IF(IGMB(L2).NE.0) GØ TØ 15
0128            XB=X(L1)
0129            YB = M(L2)*XB+B(L2)
0130            GØ TØ 17
0131      14    IF(IGMB(L2).EQ.0) GØ TØ 18
0132            XB=X(L2)
0133            YB=M(L1)*XB+B(L1)
0134            GØ TØ 17
0135      15    IF(X(L1).NE.X(L2)) GØ TØ 23
0136            IF((Y(L1)-Y(L2 ))*(Y(L1P)-Y(L2 )).LE.0) GØ TØ 21
0137            IF((Y(L1)-Y(L2P))*(Y(L1P)-Y(L2P)).LE.0) GØ TØ 21
0138            IF((Y(L2)-Y(L1 ))*(Y(L2P)-Y(L1 )).LE.0) GØ TØ 21
0139            IF((Y(L2)-Y(L1P))*(Y(L2P)-Y(L1P)).LE.0) GØ TØ 21
0140            GØ TØ 23
0141      21    WRITE(6,1003)
0142      1003  FØRMAT(' BAD PØINT (DELETED) !!!'/)
0143            CALL ØUTPUT(7)
0144            GØ TØ 22
0145      18    MM = M(L2)-M(L1)
0146            IF(MM.NE.0) GØ TØ 19
0147            IF(B(L2).NE.B(L1)) GØ TØ 23
0148            IF((X(L1)-X(L2))*(X(L1P)-X(L2)).LE.0) GØ TØ 21
0149            IF((X(L1)-X(L2P))*(X(L1P)-X(L2P)).LE.0) GØ TØ 21
0150            IF((X(L2)-X(L1))*(X(L2P)-X(L1)).LE.0) GØ TØ 21
0151            IF((X(L2)-X(L1P))*(X(L2P)-X(L1P)).LE.0) GØ TØ 21
0152            GØ TØ 23
0153      19    XB = (B(L1)-B(L2))/MM
0154            YB = M(L1)*XB+B(L1)
0155      17    IF((X(L1)-XB)*(X(L1P)-XB).GT.0) GØ TØ 23
0156            IF((X(L2)-XB)*(X(L2P)-XB).GT.0) GØ TØ 23
0157            IF((Y(L1)-YB)*(Y(L1P)-YB).GT.0) GØ TØ 23
0158            IF((Y(L2)-YB)*(Y(L2P)-YB).LE.0) GØ TØ 21
0159      23    L2 = L2 - 1
```

```
0160            GO TO 24
0161     500      IF(DDIESF.NE.0)GO TO 501
0162              DO 502 K = 1,N+1
0163              JX=BE(1)+(X(K)-RI(1))*RSF(1)+.5
0164              JY=BE(2)+(Y(K)-RI(2))*RSF(2)+.5
0165              FIELD(2*K+7,FPTR)=JX
0166              FIELD(2*K+8,FPTR)=JY
0167     502      CONTINUE
0168              FIELD(8,FPTR)=N+1
0169     501    SD = 0
0170              K = N + 1
0171            IF(IPAINT.EQ.0)GO TO 506
         C        DO 503 N = 2,K
         C        CALL SDPNT(X,Y,N,SD,THM)
         C503     CALL SDLINE(SD)
         C        N = N - 2
0172              IF(N.LT.3)GO TO 504
0173            CALL CVRDEF
0174     506      IF(NFLDS.EQ.0) GO TO 505
0175       504  GO TO 42
0176       505  CALL DETACH
0177              OPEN(UNIT=1,NAME='[300,1]IRRDEF.TMP',TYPE='OLD',FORM=
                 1 'UNFORMATTED')
0178              WRITE(1)PLOCK
0179              CLOSE(UNIT=1)
0180            CALL SETEF(53)
0181            END
```

ORIGINAL PAGE IS
OF POOR QUALITY

10.3 SUBROUTINE IRREG3

A flow chart for this subroutine can be found in volume 1, page 3-168.

10.4 SUBROUTINE CURDEF

```
HFORTRAN IV-PLUS V02-04          18:39:42    29-AUG-77          PAGE 1
CURDEF.FTN      /TR:BLOCKS/WR
0001            SUBROUTINE CURDEF
0002            INTEGER CYC(32),PTN(30),LINS(30,30),XX(30),X(30),Y(30),IG(30),
               1 YY(30),THM
0003            REAL A(30),B(30)
0004            COMMON /CURCOM/X,Y,N,THM,A,B,IG,LINS,CYC,PTN,XX,YY,
               1 IXM(2),IBUF(4096)
0005            CYC(1)=N
0006            DO 1 I = 1,N
0007            PTN(I)=I
0008            CYC(I+1)=I
0009            XX(I)=Y(I)
0010              DO 1 J = 1,N
0011    1         LINS(I,J)=0
0012            CYC(N+2)=1
0013            CALL SORT(PTN,XX,N)
0014              NM1=N-1
0015            DO 2 I = 2,NM1
0016            IN = 0
0017              IM1=I-1
0018            DO 2 J1 = 1,IM1
0019            K = PTN(J1)
0020            K1 = CYC(K)
0021            K2 = CYC(K+2)
0022            J2 = I + 1
0023    3       L = PTN(J2)
0024            IF(L.NE.K1) GO TO 4
0025            K1 = 0
0026            GO TO 5
0027    4       IF(L.NE.K2) GO TO 6
0028            K2 = 0
0029    5         IF(L.EQ.CYC(K+2))L=K
0030    8       IN = IN + 1
0031            LINS(IN,I)=L
0032    6       J2 = J2 + 1
0033            IF(J2.GT.N) GO TO 2
0034            IF(K1.NE.0)  GO TO 3
0035            IF(K2.NE.0) GO TO 3
0036    2       CONTINUE
0037            I = 1
0038    105     J = PTN(I)
0039            K = PTN(I+1)
0040            IF(Y(J).NE.Y(K))GO TO 99
0041            DO 98 L = 1,N
0042    98      LINS(L,I) = 0
0043            GO TO 101
0044    99      L = 0
0045    94        IF(LINS(L+1,I).EQ.0) GO TO 97
0046            L = L + 1
0047            IF(L.LE.N) GO TO 94
0048            GO TO 101
0049    97      IF(I.GT.N-1) GO TO 102
0050            J = 1
0051    104     K = LINS(J,I+1)
0052            IF(K.EQ.0) GO TO 102
0053            LL = 1
0054    103     IF(LINS(LL,I).EQ.K) GO TO 96
```

```
C:ORDEF,FIN      /TR:BLOCKS/WR
0099            LL = LL + 1
0096            IF(LL.LE.L) GO TO 103
0097            L = L + 1
0098            LINS(L,I)=K
0099      96    J = J + 1
0100            IF(J.LE.N) GO TO 104
0101      102   IF(L.EQ.(L/2)*2) GO TO 101
0102            LL = PTN(I)
0103            K = PTN(I+1)
0104            IF(LL.LT.K) GO TO 95
0105            J = LL
0106            LL = K
0107            K = J
0108      95    IF(LL.EQ.1.AND.K.EQ.N) LL = N
0109            L = L + 1
0110            LINS(L,I) = LL
0111      101   I = I + 1
0112            IF(I.LT.N) GO TO 105
          C     STEP 4
0113            I = 1
0114            NM1 = N-1
0115            DO 204 II = 1,NM1
0116            IF(LINS(1,II).EQ.0) GO TO 204
0117              K1 = PTN(II)
0118              K2 = PTN(II+1)
0119              IY=(Y(K1)+Y(K2))/2
0120            DO 205 J = 1,N
0121            IF(LINS(J,II).EQ.0) GO TO 206
0122              JN = J
0123              K1 = LINS(J,II)
0124              IF(IG(K1).EQ.0) GO TO 207
0125              XX(J)=X(K1)
0126              GO TO 205
0127      207     CONTINUE
0128              AA=(FLOAT(IY)-R(K1))/A(K1)
0129              XX(J)=AA
0130      205     CONTINUE
0131      204     CONTINUE
0132            IF(I.EQ.II) GO TO 203
0133            DO 202 J = 1,N
0134      202   LINS(J,I)=LINS(J,II)
0135      203   CALL SORT(LINS(1,I),XX,JN)
0136              I = I + 1
0137      204     CONTINUE
0138            N = I - 1
0139            CALL PAINT
0140            RETURN
0141            END
```

10.4 SUBROUTINE CURDEF

A flow chart for this subroutine is not available.

10.5 SUBROUTINE FAKCU3

```
MEØRTRAN IV-PLUS V02-04          12136111   29-AUG-77          PAGE 1
FAKCU3.FTN       /TR:BLØCKS/WR
0001             SUBRØUTINE FAKCU3(BLØCK)
0002             CØMMØN /CURCØM/ ID(1112),IA(37)
0003             CØMMØN /FAKCØM/RI(2),RD(2),RSF(2)
                *,IP,CURFIL
0004               INTEGER S
0005             INTEGER CURFIL
0006               INTEGER BLØCK(15)
0007        2    WRITE(6,1000)
0008        1000 FØRMAT('$USE LATEST IMAGE DISPLAY CØØRDINENTS (Y)ES/(N)Ø  > ')
0009             READ(IP,1001)IA
0010        1001 FØRMAT(37A2)
0011             IF(IA(1).EQ.'Y') GØ TØ 1
0012             IF(IA(1).NE.'N') GØ TØ 2
0013             WRITE(6,1002)(BLØCK(I),I=4,7)
0014        1002 FØRMAT('$CØØRDINATES FØR DATA BASE IMAGE ',4(I3,1X),'  > ')
0015           1 I=0
0016             READ(IP,1001)IA
0017             CALL INTFF(I,IA,74,LX1)
0018             CALL INTFF(I,IA,74,LY1)
0019             CALL INTFF(I,IA,74,LX2)
0020             CALL INTFF(I,IA,74,LY2)
0021             WRITE(6,1003)(BLØCK(I),I=8,11)
0022        1003 FØRMAT('$CØØRDINATES FØR DISPLAY IMAGE ',4(I3,1X),'  > ')
0023           1 I=0
0024             READ(IP,1001)IA
0025             CALL INTFF(I,IA,74,IX1)
0026             CALL INTFF(I,IA,74,IY1)
0027             CALL INTFF(I,IA,74,IX2)
0028             CALL INTFF(I,IA,74,IY2)
0029        3    IX1 = IAND(IX1,511)
0030             IY1 = IAND(IY1,511)
0031             IX2 = IAND(IX2,511)
0032             IY2 = IAND(IY2,511)
0033             RI(1)=IX1
0034             RI(2)=IY1
0035             RD(1)=LX1
0036             RD(2)=LY1
0037               IQ = 0
0038               IQQ = 0
0039               IF(LX1.LE.LX2) GØ TØ 4
0040               IQ = -2
0041        4      IF(LY1.LE.LY2) GØ TØ 5
0042               IQQ = -2
0043        5      CØNTINUE
0044             RSF(1)=FLØAT(LX2-LX1+1+IQ)/(IX2-IX1+1)
0045             RSF(2)=FLØAT(LY2-LY1+1+IQQ)/(IY2-IY1+1)
0046               WRITE(6,1004)LX1,LY1,LX2,LY2
0047        1004   FØRMAT(' DATA BASE CØØRDINATES >'4I5)
0048               WRITE(6,1005)IX1,IY1,IX2,IY2
0049        1005   FØRMAT(' DISPLAY CØØRDINATES  >'4I5)
0050               WRITE(6,1006)
0051        1006   FØRMAT('$PRØCEED (Y)ES/(N)Ø  > ')
0052               READ(6,1001)IA
0053               IF(IA(1).NE.'Y')GØ TØ 2
0054             RETURN
0055        1      LX1=BLØCK(4)
```

```
0056            LY1=BLOCK(5)
0057            LX2=BLOCK(6)
0058            LY2=BLOCK(7)
0059            IX1=BLOCK(8)
0060            IY1=BLOCK(9)
0061            IX2=BLOCK(10)
0062            IY2=BLOCK(11)
0063          GO TO 3
0064          END
```

10.5 SUBROUTINE FAKCU3

A flow chart for this subroutine is not available.

10.6 SUBROUTINE RDXYD3

```
FORTRAN IV-PLUS V02-04          12:30:40  29-AUG-77          PAGE 1
RDXYD3.FTN      /TR:RLOCKS/LR
0001            SUBROUTINE RDXYD3(IX,IY,ID,IDEL,CURFIL,FPTR,FIELD,N)
0002              INCLUDE '[300,3]CAMSPARAM.INC'
0003 *           PARAMETER MAXCAT=80,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=90
     *          1,MAXV=11,NPOTS=200,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *          2NOSPWD=4,NRDTWD=10
0004              INTEGER *,FPTR
0005              BYTE FIELD(2*MAXV*8,MAXFLD)
0006              INTEGER FI
0007              COMMON/FAKCOM/  PIX(2),REF(2),RSF(2)
                 * ,IP,CURFL
0008              COMMON/CURCOM/IDDD(1112),A
0009              INTEGER CURFIL
0010             INTEGER SW(8),CR(5),A(37) ,P
0011             IDEL = 0
0012             ID = 0
0013             IF(CURFIL.NE.0) GO TO 1
0014        3    CALL IRS(SW)
0015             IF(IAND("100,SW(1)),NE.0) GO TO 4
0016             IF(IAND("200,SW(1)),NE.0) GO TO 5
0017             CALL IRPOLL(P)
0018             IF(P.NE."104) GO TO 3
0019        6    IDEL = 1
0020             RETURN
0021        5    ID = 1
0022             RETURN
0023        4    CALL IRK(CR)
0024             IX = CR(2)
0025             IY = CR(4)
0026               SW(1)=04
0027               CALL IRS(SW)
0028             RETURN
0029        1      IF(CURFIL.NE.1) GO TO 2
0030             WRITE(6,2001)
0031        2001   FORMAT('ENTER DATA BASE X,Y FOR CORNER OR (DO)NE OR (DE)LETE',
                   1 '  > ')
0032             READ(6,2000)A
0033        2000 FORMAT(37A2)
0034             IF(A(1).EQ.'DO') GO TO 5
0035             IF(A(1).EQ.'DE') GO TO 6
0036             I = 0
0037             CALL IRTFF(I,A,74,JX)
0038             CALL IRTFF(I,A,74,JY)
0039        7      IX = (JX-REF(1))/RSF(1)+PIX(1)+.5
0040             IY = (JY-REF(2))/RSF(2)+PIX(2)+.5
0041             RETURN
0042        2    CONTINUE
0043               FI = FIELD(8,FPTR)
                 WRITE(6,5002)FI,*,FPTR
            5002   FORMAT(' ',415)
0044               IF(N.GE.FI+1) GO TO 5
0045               JX = FIELD(2*N+9,FPTR)
0046               JY = FIELD(2*N+10,FPTR)
0047               JX = IAND(JX,255)
0048               JY = IAND(JY,255)
            3      WRITE(6,5002)JX,JY
0049               GO TO 7
```

RDXYD3.FTN /TR:BLOCKS/WR
0050 END

C-4

10.6 SUBROUTINE RDXYD3

A flow chart for this subroutine is not available.

10.7 SUBROUTINE SDLINE

```
MEØRTRAN IV PLUS V02-04          12:37:13   29-AUG-77          PAGE 1
SDLINE.FT       /TRIBLØCKS/WR
0001            SUBRØUTINE SDLINE(SD)
0002            CØMMØN/CURCØM/X(30),Y(30),N,THM,IDM(1174),BUF
0003            INTEGER X,Y,N,THM,SD,BUF(32,50,2),BASE,MAX,A,B,D,T,I,F,XZ,YZ,
               1 MV,CØDE(16),E,PLT,SPAN
               2 ,XN,YN
0004              INTEGER Ø,SPS2
0005            DATA CØDE/1,2,3,2,3,4,5,4,5,6,7,6,7,8,1,8/
0006            XZ = X(N-1)
0007            YZ = Y(N-1)
0008            XN = X(N)
0009            YN = Y(N)
0010    1000        IF(YZ-YN)1,2,3
0011    2       IF(XZ.EQ.XN) RETURN
0012            GØ TØ 1
0013    3       K = XN
0014            XN = XZ
0015            XZ = K
0016            K = YZ
0017            YZ = YN
0018            YN = K
0019    1       BASE = YZ
0020              BASE = (YZ/2)*2
0021            SPAN = 100
0022            IF(YZ+SPAN.GE.512) SPAN=512-YZ
0023              SPS2=SPAN/2
0024            CALL IRT(THM,BASE,SPS2,BUF(1,1,1))
0025            CALL WAIT
0026              CALL IRT(THM,BASE+1,SPS2,BUF(1,1,2))
0027              CALL WAIT
0028            MAX = BASE + SPAN - 1
0029            A = XN-XZ
0030            B = YN-YZ
0031            D = A + B
0032            T = B - A
0033            I = Ø
0034            IF(B.GE.Ø) I = 2
0035            IF(D.GE.Ø) I = I + 2
0036            IF(T.GE.Ø) I = I + 2
0037            IF(A.GE.Ø) GØ TØ 90
0038            I = I + 10
0039            GØ TØ 91
0040    90      I = 8 - I
0041    91      A = IABS(A)
0042            B = IABS(B)
0043            F = A + B
0044            D = B - A
0045            T = B
0046            IF(D.LT.Ø) GØ TØ 92
0047            T = A
0048            D = -D
0049    92      E = Ø
0050            MV = CØDE(I-1)
0051            I = CØDE(I)
0052    93      A = D + E
0053            B = T+E+A
0054            IF(B.LT.Ø) GØ TØ 94
```

```
0055              E = A
0056              F = F - 2
0057              PLT = I
0058              GØ TØ 95
0059        94    E = E + T
0060              F = F - 1
0061              PLT = MV
0062        95      CØNTINUE
0063              IF(PLT.GT.4) GØ TØ 96
0064              IF(PLT.LE.1) GØ TØ 97
0065              XZ = XZ + 1
0066              GØ TØ 97
0067        96    IF(PLT.LT.6) GØ TØ 97
0068              XZ = XZ - 1
0069        97    IF(PLT-3)98,99,100
0070        98    YZ = YZ + 1
0071              GØ TØ 99
0072       100    IF(PLT.EQ.8) GØ TØ 98
0073              IF(PLT.GT.6) GØ TØ 99
0074              YZ = YZ-1
0075        99    IF(YZ.LE.MAX) GØ TØ 101
0076              CALL IWT(THM,BASE,SPS2,BUF(1,1,1))
0077              CALL WAIT
0078                CALL IWT(THM,BASE+1,SPS2,BUF(1,1,2))
0079                CALL WAIT
0080              SPAN = 100
0081              IF(YZ+SPAN.GE.512) SPAN=512-YZ
0082              BASE = YZ
0083              MAX = BASE + SPAN - 1
0084                SPS2=SPAN/2
0085              CALL IRT(THM,BASE,SPS2,BUF(1,1,1))
0086              CALL WAIT
0087                CALL IRT(THM,BASE+1,SPS2,BUF(1,1,2))
0088                CALL WAIT
0089       101      L = YZ-BASE+1
0090                J = 2-L+(L/2)*2
0091                L = (L+1)/2
0092                CALL BITSET(XZ,BUF(1,L,J),SD)
0093       103    IF(F.GT.0) GØ TØ 97
0094              CALL IWT(THM,BASE,SPS2,BUF(1,1,1))
0095              CALL WAIT
0096                CALL IWT(THM,BASE+1,SPS2,BUF(1,1,2))
0097                CALL WAIT
0098              RETURN
0099              END
```

10.7 SUBROUTINE SDLINE

A flow chart for this subroutine is not available.

10.8 SUBROUTINE SDPNT

```
HFORTRAN IV-PLUS V02-04                    12:37:37    29-AUG-77          PAGE 1
SDPNT.FTN          /TR:BLOCKS/WR
0001               SUBROUTINE SDPNT(X,Y,N,SD,THM)
0002               INTEGER X(1),Y(1),N,SD,THM,BUF(32)
0003                 DO 2 I = 1,2
0004               CALL IRT(THM,Y(N)+I-1,1,BUF)
0005               CALL WAIT
0006               CALL BITSET(X(N),BUF,SD)
0007                 CALL BITSET(X(N)+1,BUF,SD)
0008     2         CALL IWT(THM,Y(N)+I-1,1,BUF)
0009               CALL WAIT
0010               RETURN
0011               END
```

10.8 SUBROUTINE SDPNT

A flow chart for this subroutine is not available.

10.9 SUBROUTINE SWCLR

```
MFORTRAN IV-PLUS V02-04          12137158    29-AUG-77          PAGE 1
SWCLR.FTN       /TRIBLOCKS/WR
0001            SUBROUTINE SWCLR
0002            INTEGER SW(8)
0003            CALL IRS(SW)
0004            SW(1)=192
0005            CALL IWS(SW)
0006            SW(1)=1
0007            SW(2)=256
0008            SW(3)=0
0009            SW(4)=256
0010            SW(5)=0
0011            CALL IWK(SW)
0012            RETURN
0013            END
```

10.9 SUBROUTINE SWCLR

A flow chart for this subroutine is not available.

10.10 SUBROUTINE SORT

```
MFORTRAN IV+PLUS V02-04                    12:38:14    29-AUG-77          PAGE 1
SORT.FTN          /TRACEBACKS/WR
0001              SUBROUTINE SORT(S,K,N)
0002              INTEGER S(N),K(N),N,SS,KK,SSS,KKK
0003              NM1 = N-1
0004              DO 1 I = 1,NM1
0005              IP1=I+1
0006              SS = S(I)
0007              KK = K(I)
0008              DO 2 J = IP1,N
0009              IF(K(J).GE.KK) GO TO 2
0010              KKK = KK
0011              KK = K(J)
0012              K(J)=KKK
0013              SSS = SS
0014              SS = S(J)
0015              S(J) = SSS
0016         2    CONTINUE
0017                S(I)=SS
0018                K(I)=KK
0019         1    CONTINUE
0020              RETURN
0021              END
```

ORIGINAL PAGE IS
OF POOR QUALITY

10.10 SUBROUTINE SORT

A flow chart for this subroutine is not available.

10.11 SUBROUTINE PAINT

```
HFPHTRAN IV+PLUS V02-04              12:38:32   29-AUG-77          PAGE 1
PAINT.FTN        /TRIBLOCKS/WR
0001            SUBROUTINE PAINT
0002              INTEGER YN,XM(2),C,DD,BASE,YLST,SPAN,SPS2
0003              INTEGER THM
0004              INTEGER X2
0005              COMMON /CURCOM/X,Y,M,THM,N,B,IG,LINS,CYC,PTN,XX,YY,XM,BUF
0006              INTEGER CYC(32),PTN(30),LINS(30,30),XX(30),X(30),Y(30),IG(30),
                 1 YY(30)
0007              REAL M(30),B(30)
0008              INTEGER BUF(32,64,2)
0009             K = PTN(1)
0010             YLST=Y(K)
0011             BASE = YLST
0012               BASE = (YLST/2)*2
0013             SPAN = 100
0014             IF(BASE+SPAN.GE.512) SPAN = 512-BASE
0015               SPS2=SPAN/2
0016             CALL IRT(THM,BASE,SPS2,BUF(1,1,1))
0017             CALL WAIT
0018               CALL IRT(THM,BASE+1,SPS2,BUF(1,1,2))
0019             CALL WAIT
0020             MAX = BASE+SPAN-1
0021             I = 1
0022             II = 1
0023    3        K = PTN(II)
0024    5        L = PTN(II+1)
0025             IF(Y(K).NE.Y(L)) GO TO 100
0026             II = II + 1
0027               IF(II.LT.30) GO TO 5
0028               GO TO 201
0029    100      YN = Y(K)
0030    1        J = 1
0031    11       K = 1
0032    111      KK = 2*(J-1)+K
0033             L = LINS(KK,I)
0034             IF(L.EQ.0) GO TO 102
0035             IF(X(L).EQ.0) GO TO 103
0036             XM(K)=X(L)
0037             GO TO 105
0038    103      IF(M(L).NE.0) GO TO 104
0039             XM(K)=X(L)
0040             IF(XM(K).GT.X(L+1)) XM(K)=X(L+1)
0041             GO TO 105
0042    104      XM(K)=(YN-B(L))/M(L)
0043    105      K = K + 1
0044             IF(K.LE.2) GO TO 111
0045             X = XM(1)
0046             L = XM(2)
0047             D = YN-BASE+1
0048               DD = 2 * D + (D/2)*2
0049               C=(D+1)/2
0050             DO 106 X2 = X,L
0051    106      CALL BITSET(X2,BUF(1,D,DD),1)
0052             J = J + 1
0053             IF(J.LE.30) GO TO 1
0054    102      YN = YN + 1
0055             IF(YN.LE.MAX) GO TO 107
```

ORIGINAL PAGE IS
OF POOR QUALITY

PAINT.FTN /TR:BLOCKS/WR

```
0056            CALL IWT(THM,BASE,SPS2,BUF(1,1,1))
0057            CALL WAIT
0058              CALL IWT(THM,BASE+1,SPS2,BUF(1,1,2))
0059            CALL WAIT
0060              SPAN = 100
0061            IF(YN+SPAN.GE.512) SPAN=512-YN
0062            BASE = YN
0063            MAX = YN+SPAN-1
0064              SPS2=SPAN/2
0065            CALL IRT(THM,BASE,SPS2,BUF(1,1,1))
0066            CALL WAIT
0067              CALL IRT(THM,BASE+1,SPS2,BUF(1,1,2))
0068            CALL WAIT
0069     107    K = PTN(II+1)
0070            IF(YN.LE.Y(K)) GO TO 4
0071     200    I = I + 1
0072            II = II + 1
0073            IF(I.LE.N) GO TO 3
0074            CALL IWT(THM,BASE,SPS2,BUF(1,1,1))
0075            CALL WAIT
0076              CALL IWT(THM,BASE+1,SPS2,BUF(1,1,2))
0077            CALL WAIT
0078              RETURN
0079            END
```

10.11 SUBROUTINE PAINT

A flow chart for this subroutine is not available.

10.12 SUBROUTINE FLDNAM

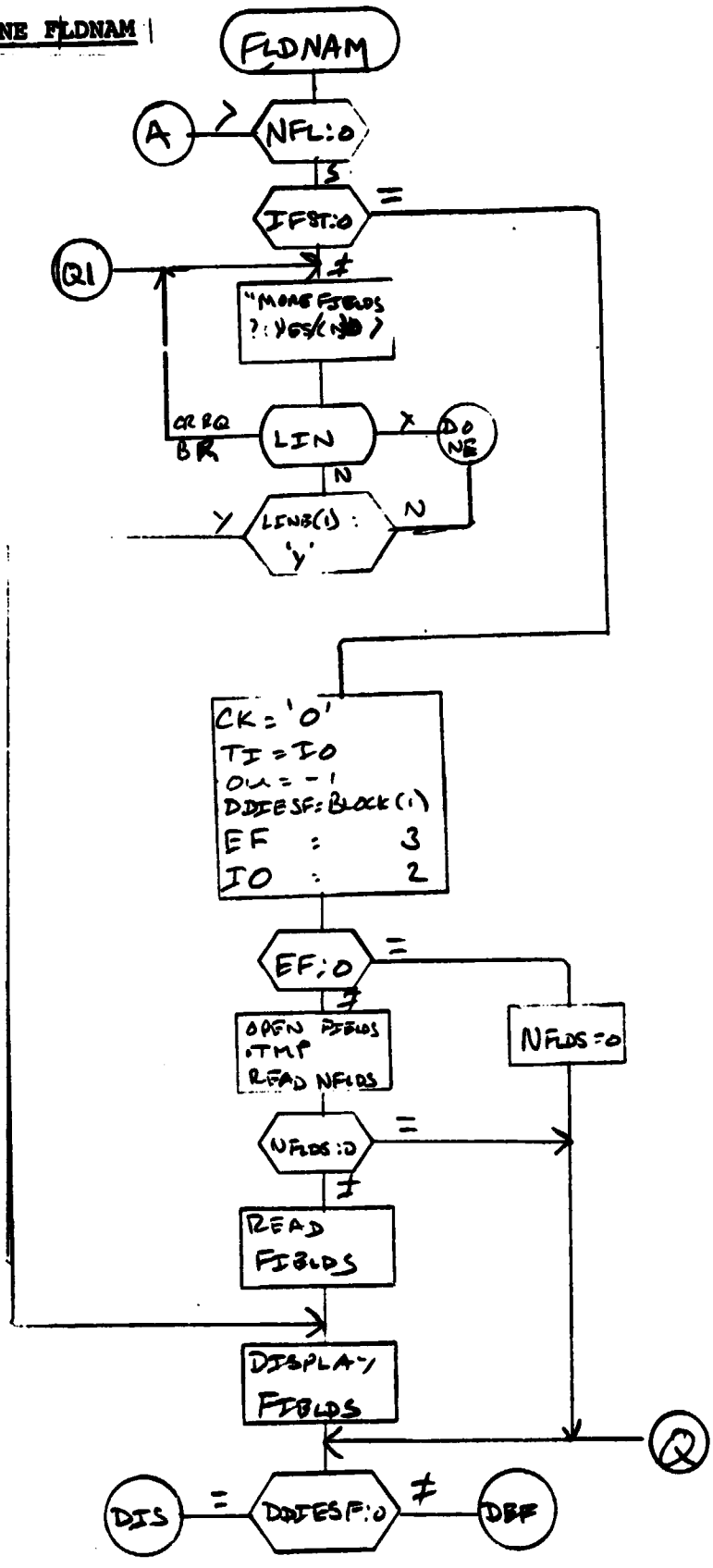
```
HFØRTRAN IV-PLUS V02-04          12:39:00   29-AUG-77          PAGE 1
FLDNAM.FTN        /TR:BLØCKS/WR
0001              SUBRØUTINE FLDNAM(MFLDS,FPTR,NFL,CURDEF,DDIESF,IFST,
                  1 FIELD,I0,NFLDS,BLØCK)
0002              IMPLICIT INTEGER (A - Z )
0003              INCLUDE '[300,3]CAMSPARAM.INC'
0004 *           PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
     *          1,MAXV=11,NDØTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
     *          2NØSPWD=6,NØDTWD=10
0005              BYTE FIELD(10+2*(MAXV-1),MAXFLD),LINE(74)
0006              BYTE FLLST(MAXFLD)
0007              INTEGER BLØCK(15),DUD0(2)
0008              BYTE CK,U0(2)
0009              DATA U0/'U','0'/
0010              DATA DUD0/'DU','D0'/
0011              IF(NFL.GT.0)GØ TØ15
0012              IF(IFST.EQ.0) GØ TØ 2
0013     3        TI = I0
0014              WRITE(I0,1000)
0015     1000     FØRMAT('$MØRE FIELDS ?  (Y)ES/(N)Ø  > ')
0016              CALL LIN(I0,LINE,TI,IF)
0017              GØ TØ (99,3,3,3,3,4),IF
0018     4        IF(LINE(1).EQ.'N') GØT Ø 99
0019              IF(LINE(1).NE.'Y') GØ TØ 3
0020              GØ TØ 5
0021     2        CØNTINUE
0022              CK='C'
0023              DDIESF=BLØCK(1)
0024              I0 = BLØCK(2)
0025              TI = I0
0026              EF=BLØCK(3)
0027              ØU = -1
0028              IF(EF.NE.0)GØ TØ 29
0029              NFLDS = 0
0030              GØ TØ 30
0031     29       ØPEN(UNIT=1,NAME='[300,1]FIELDS.TMP',TYPE='ØLD',
                  1 FØRM='UNFØRMATTED')
0032              READ(1)NFLDS
0033              NFLDSF=NFLDS
0034              IF(NFLDS.EQ.0) GØ TØ 31
0035              DØ 6 I = 1,NFLDS
0036     6        READ(1)(FIELD(J,I),J=1,2*(MAXV-1)+10)
0037     31       CLØSE(UNIT=1)
0038     5        CØNTINUE
0039              IF(NFLDS.EQ.0)GØ TØ 30
0040              WRITE(I0,1001)(I,DUD0(FIELD(7,I)+3),(FIELD(J,I),J=1,6),I=
                  1 1,NFLDS)
0041     30       IF(DDIESF.EQ.0) GØ TØ 7
0042              IF(NFLDS.EQ.0)GØ TØ 99
0043              CURDEF = 1
0044     10       WRITE(I0,1002)
0045     1002     FØRMAT(' INPUT CATAGØRY NAME ØR FIELD NUMBERS TØ BE DISPLAYED.',
                  1 '  D(Ø)/D(U)  > '/)
0046     1001     FØRMAT(' CURRENT FIELDS ARE:'/(4(' (',I2,') ',A2,2X,6A1)))
0047              NFL = 0
0048              CALL LIN(I0,LINE,TI,IF)
0049              GØ TØ (99,8,10,10,10,11),IF
0050     8        IF(IFST.EQ.0)GØ TØ 5
```

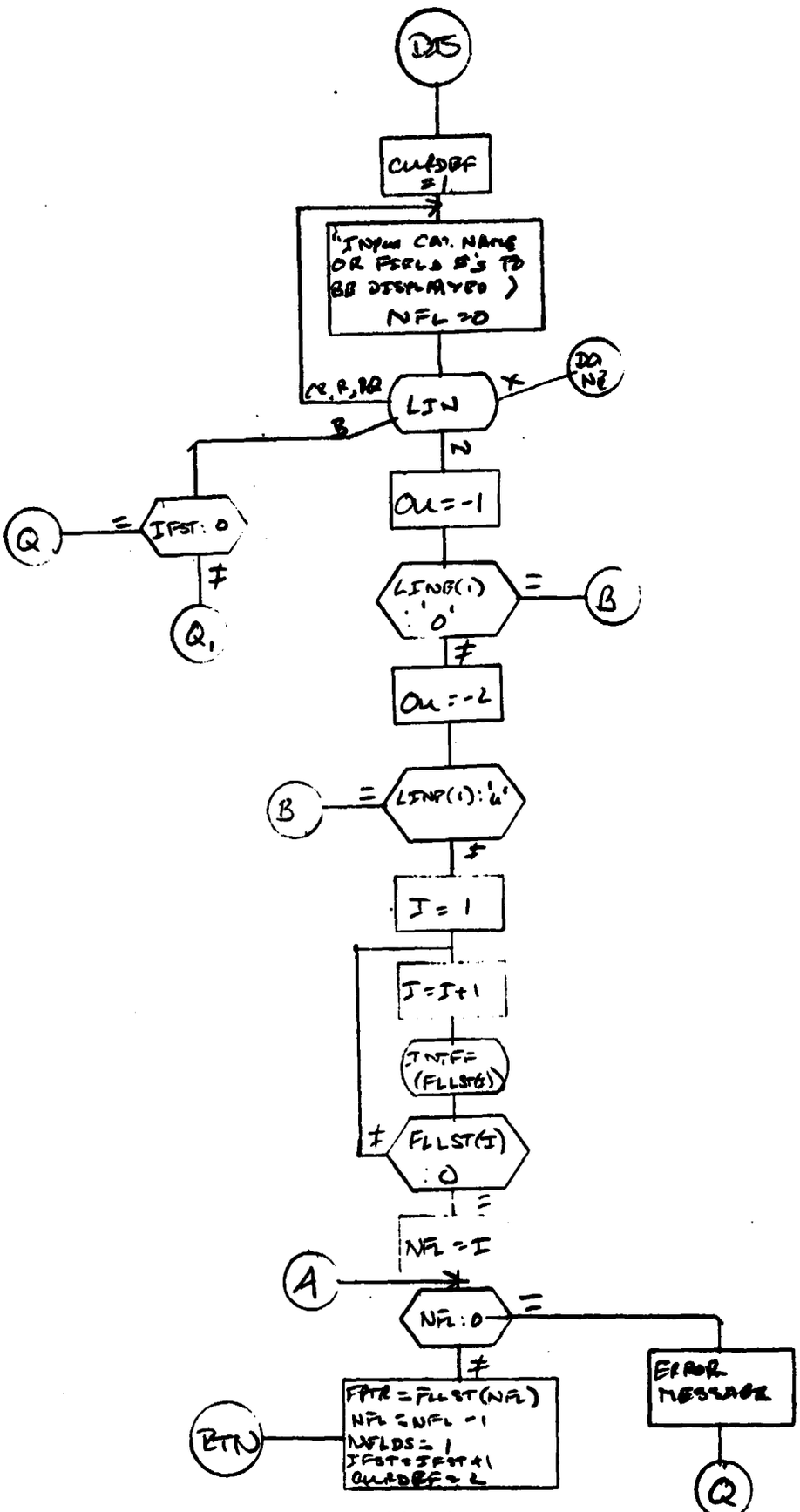

ORIGINAL PAGE IS
OF POOR QUALITY

```
0051                GO TO 3
0052     11         OU = -1
0053                IFST = 0
0054                IF(LINE(1).EQ.'0') GO TO 12
0055                OU = -2
0056                IF(LINE(1).EQ.'U')GO TO 12
0057                IP = 0
0058                DO 13 I = 1,NFLDS
0059                CALL INTFF(IP,LINE,74,IQ)
0060                FLLST(I)=IQ
0061     13         IF(FLLST(I).EQ.0)GO TO 14
0062     14         NFL = I-1
0063                GO TO 15
0064     12         NFL = 0
0065                DO 16 I = 1,NFLDS
0066                IF(FIELD(7,I).NE.0U)GO TO 16
0067                NFL = NFL + 1
0068                FLLST(NFL)=I
0069     16         CONTINUE
0070     15         IF(NFL.EQ.0) GO TO 1234
0071                FPTR=FLLST(NFL)
0072                IF(FPTR.EQ.0) GO TO 3
0073                NFL =NFL - 1
0074                NFLDS = 1
0075                IFST=IFST+1
0076                CURDEF = 2
0077                RETURN
0078     1234       WRITE(10,1235)
0079     1235       FORMAT(' NO FIELDS SELECTED, TRY AGAIN...',///)
0080                GO TO 5
0081     7          IF(NFLDS.LT.MAXFLD)GO TO 201
0082                WRITE(10,1003)
0083     1003       FORMAT(' TOO MANY FIELDS,  DELETE ONE FIRST,',///)
0084                GO TO 99
0085     201        DO 292 I = 1,6
0086     292        FIELD(I,NFLDS+1)=' '
0087     17         WRITE(10,1004)(FIELD(I,NFLDS+1),I=1,6)
0088     1004       FORMAT('$INPUT SIX CHARACTER NAME FOR NEW FIELD  ',6A1,'  > ')
0089                CALL LIN(10,LINE,TI,IF)
0090                GO TO (99,17,17,17,20,18),IF
0091     18         DO 19 I = 1,6
0092     19         FIELD(I,NFLDS+1)=LINE(I)
0093     20         WRITE(10,1005)UO(0U+3)
0094     1005       FORMAT('$INPUT CATAGORY OF NEW FIELD  D(0)/D(U)  ',A1,'  > ')
0095                CALL LIN(10,LINE,TI,IF)
0096                GOTO(99,17,20,20,22,21),IF
0097     21         0U= -1
0098                IF(LINE(1).EQ.'0') GO TO 22
0099                0U = -2
0100                IF(LINE(1).NE.'U') GO TO 20
0101     22         FIELD(7,NFLDS+1)=0U
0102                FIELD(8,NFLDS+1)=0
0103     23         WRITE(10,1006)(FIELD(I,NFLDS+1),I=1,6),DUDO(FIELD(7,NFLDS+1)
                    1 + 3)
0104     1006       FORMAT(' FIELD NAME = ',6A1/' CATAGORY = ',A2/'$CONTIN',
                    1 'UE (Y)ES/(N)O  > ')
```

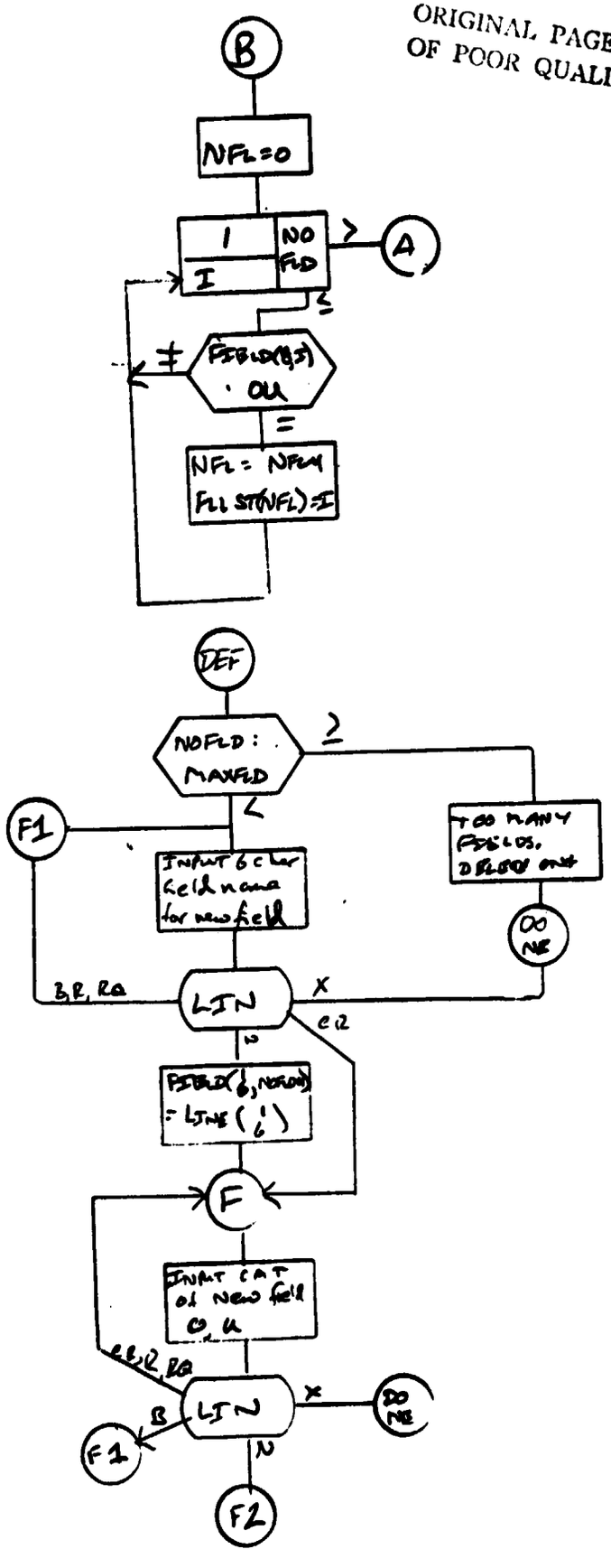
```
FØRTRAN IV-PLUS V02-04          12:39:00    29-AUG-77          PAGE 3
FLDNAM,FTN      /IR:BLØCKS/WR
0105            CALL LIN(I0,LINE,TI,IF)
0106            G0 T0 (99,20,23,23,23,241),IF
0107    241     IF(LINE(1),EQ,'N')G0 T0 17
0108            IF(LINE(1),NE,'Y') G0 T0 23
0109    24      WRITE(I0,1007)CK
0110    1007    FØRMAT('$DEFINE FIELD VERTICES BY (C)URSØR ØR (K)EYBØARD?  '
                1 ,A1,'  > ')
0111            CALL LIN(I0,LINE,TI,IF)
0112            G0 T0 (99,23,24,24,204,25),IF
0113    204     LINE(1)=CK
0114    25      CURDEF = 0
0115            CK = LINE(1)
0116            IF(LINE(1),EQ,'C') G0 T0 26
0117            IF(LINE(1),NE,'K') G0 T0 24
0118            CURDEF = 1
0119    26      NFLDS = 1
0120            IFST=1
0121            NFLDS = NFLDS+1
0122            FPTR = NFLDS
0123            RETURN
0124    99      NFLDS = 0
0125            BL0CK(2)=0
0126            IF(NFLDS,EQ,NFLDSF) G0 T0 28
0127            0PEN(UNIT=1,NAME='[300,1]FIELDS.TMP',TYPE='UNKN0WN',
                1 FØRM='UNFØRMATTED')
0128            WRITE(1)NFLDS
0129            IF(NFLDS,EQ,0) G0 T0 98
0130            D0 27 I = 1,NFLDS
0131    27      WRITE(1)(FIELD(J,I),J=1,10+2*(MAXV-1))
0132    98      CL0SE(UNIT=1)
0133            EF = 1
0134            BL0CK(2)=1
0135    28      BL0CK(1)=NFLDS
0136            BL0CK(3)=EF
0137            RETURN
0138            END
```

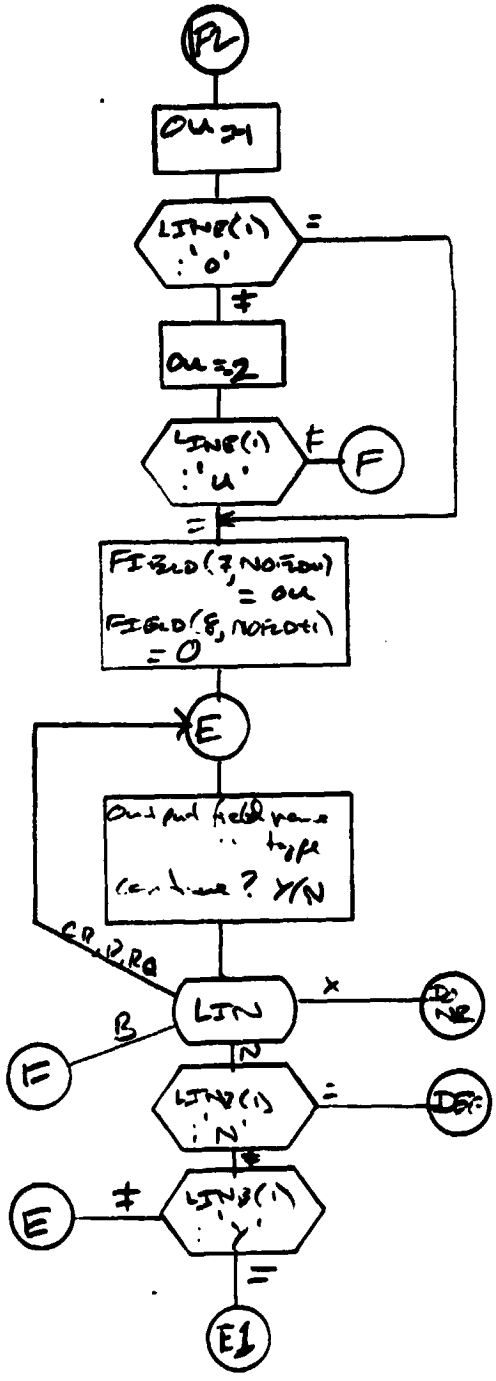
10.12 SUBROUTINE FLDNAM

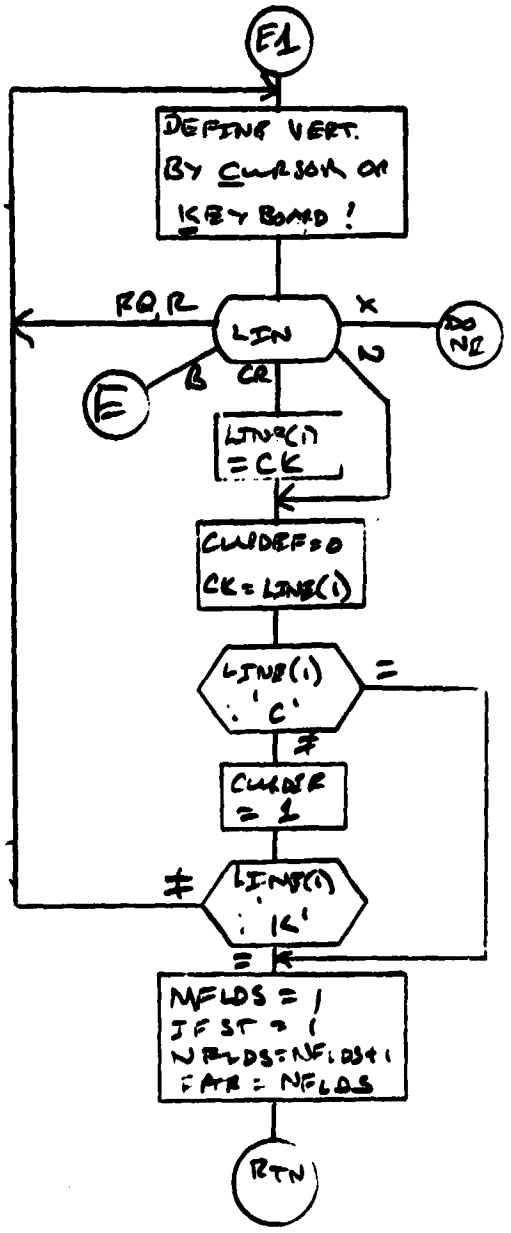



D25
CURSOR = 1
"INPUT CAT. NAME OR FIELD #'S TO BE DISPLAYED)
NFL = 0
CR, P, RQ
LIN
X
DO N?
B
N
Q
=
IFST: 0
≠
Q1
CH = -1
LINE(1) : '0'
=
B
≠
CH = -2
B
=
LINP(1) : '4'
≠
J = 1
J = J + 1
INTFF (FLLST6)
≠
FLLST(J) : 0
=
NFL = I
A
NFL : 0
=
≠
ERROR MESSAGE
Q
FPTR = FLLST(NFL)
NFL = NFL - 1
NFLDS = 1
IFST = IFST + 1
CURSOR = 6
BTN

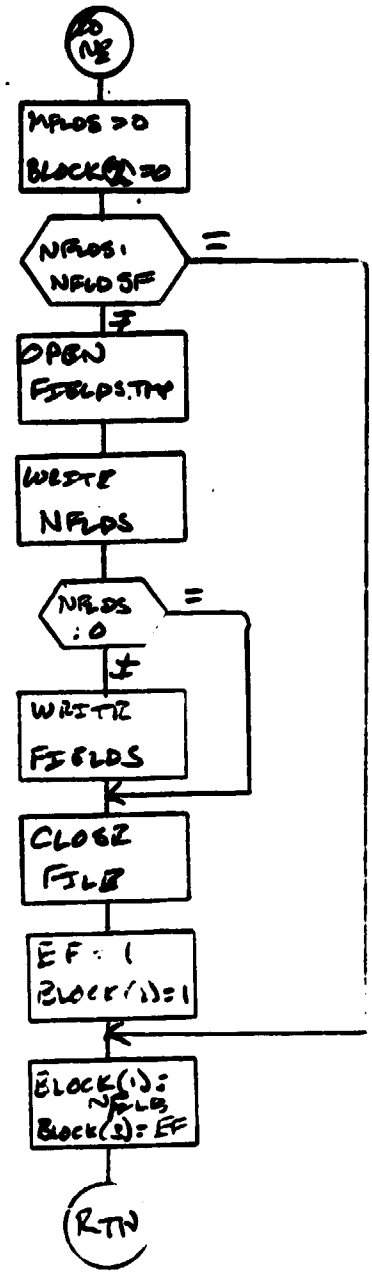
ORIGINAL PAGE IS
OF POOR QUALITY




P2
OU=1
LINE(1) = 'O'
≠
OU=2
LINE(1) = 'U'
≠
F
=
FIELD(7,NO.FLDS) = OU
FIELD(8,NO.FLDS) = 0
E
CR, 2, RQ
LIN
X
B
N
LIN(1) = 'N'
=
E
≠
LIN(1) = 'Y'
=
E1



ORIGINAL PAGE IS
OF POOR QUALITY


NFLDS > 0
BLOCK(2) = 0
NFLDS = NFLDSF
=
≠
OPEN FIELDS.TMP
WRITE NFLDS
NFLDS = 0
=
≠
WRITE FIELDS
CLOSE FILE
EF = 1
BLOCK(1) = 1
BLOCK(1) = NFLD
BLOCK(2) = EF
RTN

11. DOT GROUP CROSSHAIR OVERLAY DOTOVR

```
HFORTRAN IV-PLUS V02-04          12:39:31   29-AUG-77          PAGE 1
DOTOVR.FTN      /TR:BLOCKS/WR
      C
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      C                                                             C
      C   ROTUTINE TO OVERLAY IMAGE WITH '+' CROSSHAIR MARKS       C
      C   - DOTOVR.FTN(SUBROUINE)                                   C
      C                                                             C
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      C
      C       TITLE ..............  DOT CROSSHAIR OVERLAY
      C
      C       AUTHOR ..............  RODOLFO M. RODRIGUEZ - LEC
      C
      C       SYMBOLIC NAME .....  DOTOVR
      C
      C       DATE ................  APRIL 1977
      C
      C       LOCATION ..........  NASA/JSC
      C
      C       SOURCE LANGUAGE ...  FORTRAN IV - PLUS
      C
      C       COMPUTER TYPE .....  PDP11/45
      C
      C       OPERATING SYSTEM ..  RSX11D
      C
      C       PROJECT ............  CAMS/1-100
      C
      C .....    ....    .....    .....    .....    .....    .
      C
      C
0001            INCLUDE '[300,3]CAMSCOMON.INC'
0002  -         INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0003  -         PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      -        1,MAXV=11,NDOTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      -        2NOSPWD=6,NODTWD=10
0004  -         EQUIVALENCE (C1,ACDATE),(C2,ISEG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0005  -         INTEGER C1(469),C2(256),C3(71),C4(348),C5(629)
      -   C*
0006  -         INTEGER ACDATE,SUBCAT,SUBPOP,CATKNT,CATTH
0007  -         BYTE CHNVEC,NOCHAN,NOSUB,DOTCAT,DOTCLU
0008  -         COMMON/COM1/ACDATE(2,MAXACC),CHNVEC(MAXCHN,MAXACC),NOCHAN,NOSUB,
      -        1SUBCAT(MAXSUB),SUBPOP(MAXSUB),CATKNT(MAXCAT),CATTH(MAXCAT),NODO,
      -        2NOOU,NOTH,DOTCAT(NDOTS),DOTCLU(NDOTS)
      -   C*
0009  -         INTEGER ADATES,SUNAZ,ANALST,FLDDAY,DOTDAY,PDATE1,TDATE1
0010  -         INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RANDOM,GRID
0011  -         BYTE DELFLG,NOACQ,SOILOR,SUNEL,NSTART,NTYPE1,ALP,ALPO
0012  -         BYTE PCTCT,PCTCTO,VAR,VARO,DLABEL,TYPE
0013  -         COMMON/COM2/ISEG,DELFLG,NOACQ,ADATES(2,MAXACD),SOILOR(MAXACD),
      -        1SUNEL(MAXACD),SUNAZ(MAXACD),IMDATE(2),ANALST(5),FLDDAY(2),
      -        2DOTDAY(2),NSTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      -        3PDATE3(2),TDATE3(2),NOCAT,CATNAM(MAXCAT),ALP(MAXCAT),ALPO,
      -        4          PCTCT(MAXCAT),PCTCTO,VAR(MAXCAT),VARO
      -   C*
0014  -         INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      -        1UFLAG4
0015  -         INTEGER PFLAG,DSKMNT
```

```
0016 *        COMMON/COM3/PTL,Q,DSKMNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
     *       1,UFLAG2,UFLAG3,UFLAG4,NEWLAB(MAXSUB)
     *  C*
0017 *        INTEGER TX1,TY1,TX2,TY2,ACDISP,Q,B,DTWIND,DOTARY,GMIN,GMAX,FUL
0018 *        INTEGER SPWIND,CLAWND,CLUWND
0019 *        COMMON/COM4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACDISP(2),III(4),Q(4),
     *       1B(4),DTWIND(5,NODTWD),SPWIND(5,NOSPWD),IMWIND(4),NUMDOT,
     *       2DOTARY(NDOTS),GMIN,GMAX,FUL(2,7),CLAWND(8),CLUWND(8)
0020 *        COMMON/COM5/DISKID,RANDOM(NDOTS),GRID(NDOTS),DLABEL(NDOTS),
     *       1TYPE(NDOTS),RECLOC
      CC
      C
      C
0021          DIMENSION W(74),T(8),DAY(9),TXX(4),JX(8),NRA(1024),NRB(1024),
             1KA(2),IMS(8),ISX(4),IUY(4),IAX(4),MDATE(9)
0022          DIMENSION DOTORY(NDOTS)
      C
0023          BYTE W,T,DAY,NRA,NRB,IMS,KA,MDATE
      C         INTEGER*2 MX,MY
0024          REAL MX,MY
0025          INTEGER DOTORY
      C
      C
0026          EQUIVALENCE (TX1,TXX),(KK,KA(1)),(ISX,IMWIND),
             1  (IAX,CLUWND(5)),(IAX,CLAWND(5))
      C
0027          DATA IMS/"177,"277,"337,"357,"367,"373,"375,"376/
      C
0028          DATA MDATE/1H1,1H5,1H-,1HM,1HA,1HY,1H-,1H7,1H7/
0029          MVERS=2H01
      C
      C     CALL INIT1
0030          CALL ATTACH
0031    5     CALL OUTPUT(27,12)
0032        1 WRITE(6,1000)MVERS,MDATE
0033     1000 FORMAT(20X,'**** DOT CROSSHAIR OVERLAY - ',A2,'/',2A1,'  ****',//)
      C
      C
0034          CALL DATE(DAY)
0035          CALL TIME(T)
      C
0036          WRITE(6,1001)T,DAY
0037     1001 FORMAT(/53X,8A1,/,53X,9A1/)
      C
      C     ..... CONSTANTS .....
      C
      C                          START TIME
0038          TI=SECONDS(0.0)
      C
      C         WRITE(5,818) CLAWND,CLUWND
0039     818  FORMAT(/1X,'CLAWND,CLUWND',//(1X,8I8)/)
0040     410  FORMAT(6X,'DOTARY AFTER 49 LOOP',/,(1X,1014)/)
0041     420  FORMAT(1X,'DOTORY AFTER 49 LOOP',/,(1X,1014)/)
      C
      C   9 CONTINUE
      C   GET GRID COORDINATES AND MAGNIFICATION
```

```
C
C       IMAGE = ISX
C
C       CLUSTER = IUX
C
C       CLASSIF. = IAX
C
C      QUERY USER FOR GRID COORDINATES.
C
0042          WRITE(5,1002)ISX
0043     1002 FORMAT(10X,'SELECT GRID COORDINATES!',/,10X,
     1' = (S)TANDARD      = ',416)
C
0044          WRITE(6,1003)IUX
0045     1003 FORMAT(10X,' = CL(U)STER        = ',416)
C
0046          WRITE(6,1004)IAX
0047     1004 FORMAT(10X,' = CL(A)SSIFICATION = ',416,/,10X,' = OR INPUT ',
     1/,10X,'UPPER LEFT, LOWER RIGHT IMAGE DISPLAY COORDINATES ',',/,
     213X,'FOLLOWED BY ',',/,
     310X,'UPPER LEFT, LOWER RIGHT DATA BASE IMAGE COORDINATES ',',/,
     4'$    > ')
C
0048          CALL OUTPUT(7)
0049        4 READ(6,2000)W
0050     2000 FORMAT(74A1)
0051          IF(W(1).EQ.'S')GO TO 10
0052          IF(W(1).EQ.'U')GO TO 20
0053          IF(W(1).EQ.'A')GO TO 30
0054          IF(W(1).EQ.'X')GO TO 3000
0055          IF(W(1).EQ.'R')GO TO 5
0056          IF(W(1).LT.'0')GO TO 40
0057          IF(W(1).GT.'9')GO TO 40
0058          IP=0
0059          DO 9 I=1,8
0060          CALL ITFF(IP,W,74,JX(I))
C             WRITE(5,1234) JX(I)
0061     1234    FORMAT(6X,'JX(I)',I6/)
0062          IF(JX(I).LT.0)GO TO 40
0063          IF(JX(I).GT.521)GO TO 40
0064        9 CONTINUE
0065          GO TO 50
C
0066       10 DO 12 I=1,4
0067          JX(I)=IWIND(I)
0068       12 CONTINUE
0069          J=5
0070          DO 14 I=1,4
0071          JX(J)=TXX(I)
0072          J=J+1
0073       14 CONTINUE
0074          GO TO 50
C
0075       20 DO 22 I=1,8
0076          IF(I .LE. 4)JX(I)=CLUWND(I+4)
0077          IF(I .LE. 4) GO TO 22
```

```
0078              JX(I)=CLUWND(I-4)
0079           22 CONTINUE
0080              GO TO 50
               C
0081           30 DO 32 I=1,8
0082              IF(I .LE. 4) JX(I)=CLAWND(I+4)
0083              IF(I .LE. 4) GO TO 32
0084              JX(I)=CLAWND(I-4)
0085           32 CONTINUE
0086              GO TO 50
               C
0087           40 WRITE(6,1005)
0088         1005 FORMAT(' ***** INVALID INPUT ***** > ')
0089              CALL OUTPUT(7,7)
0090              GO TO 6
               C
               C    COMPUTE MAGNIFICATION FACTORS ...
               C
0091           50 CONTINUE
               C     WRITE(5,417) (JX(I),I=1,8)
0092          417  FORMAT(/6X,'JX',/,(1X,8I8)/)
               C
0093              IF(JX(7).NE.0)GO TO 51
0094                JX(5)=1
0095                JX(6)=1
0096                JX(7)=197
0097                JX(8)=118
0098           51 CONTINUE
0099              IA=JX(3)-JX(1)
0100              IB=JX(7)-JX(5)
               C     WRITE(5,1237) IA,IB
0101         1237  FORMAT(6X,'IA,IB',2I8/)
0102              MX=(FLOAT(IA+1))/(FLOAT(IB+1))
               C     MX=MX*100
               C     WRITE(5,1236) JX(3),JX(1),JX(7),JX(5)
0103         1236  FORMAT(/6X,'JX(3),...,JX(5)',4I8/)
0104              IC=JX(4)-JX(2)
0105              ID=JX(8)-JX(6)
0106              MY=(FLOAT(IC+1))/(FLOAT(ID+1))
               C     MY=MY*100
               C     WRITE(5,1235) MX,MY
0107         1235  FORMAT(/6X,'MX,MY',2X,F6.2,6X,F6.2/)
               C
0108               IF(MX.LT..1)GO TO 40
0109               IF(MY.LT..1)GO TO 40
               C
               C
               C    SET CROSSHAIR MARK SIZE ...
               C
0110              MX2=3
0111              MXY=3
0112              MXH=1
               C
               C
               C    SET THEME NUMBER ...
               C
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
0113        52 CONTINUE
0114           NT=8
0115             WRITE(6,1020)
0116      1020 FORMAT(1H0,/)
0117           WRITE(6,1006)NT
0118      1006 FORMAT('$        THEME NUMBER =',I2,' >')
           C
0119           CALL ØUTPUT(7)
0120        55 READ(6,2000)W
0121           CALL FRØNT(W,74)
0122           IF(W(1).EQ.' ')GØ TØ 70
0123           IF(W(1).EQ.'X')GØ TØ 3000
0124           IF(W(1).EQ.'R')GØ TØ 5
0125           IF(W(1).LT.'1')GØ TØ 60
0126           IF(W(1).GT.'8')GØ TØ 60
0127           IP=0
0128           CALL INTFF(IP,W,74,NT)
0129           GØ TØ 70
           C
0130        60 WRITE(6,1005)
0131           CALL ØUTPUT(7,7)
0132           GØ TØ 55
           C
           C
           C
           C   SET DØT SUBSET ØPTION BY UTILIZING DØT SUBSET RØUTINES ...
           C
0133        70 CØNTINUE
0134             IO=6
0135           CALL DØTIN(IO,II)
           C
           C       ØRDER SELECTED DØTS BY LINE.
           C
0136             DØ 69 I=1,NDØTS
0137             DØTØRY(I)=DØTARY(I)
0138      69     CØNTINUE
           C       WRITE(5,419) (DØTARY(I),I=1,NDØTS)
           C       WRITE(5,420) (DØTØRY(I),I=1,NDØTS)
0139      73     KFLG=0
0140             DØ 72 I=2,NUMDØT
0141             IF(DØTØRY(I-1) .LE. DØTØRY(I)) GØ TØ 72
0142             KDØT=DØTØRY(I)
0143             DØTØRY(I)=DØTØRY(I-1)
0144             DØTØRY(I-1)=KDØT
0145             KFLG=1
0146      72     CØNTINUE
0147             IF(KFLG .EQ. 1) GØ TØ 73
           C       WRITE(5,418) (DØTØRY(I),I=1,NDØTS)
0148      418    FØRMAT(6X,'DØTØRY AFTER SØRT',/,(1X,10I4))
           C
           C       DØTS ARE NØW ØRDERED BY LINE
           C
           CCC
           C       NUMDØT=8
           C       DØTARY(1)=1
           C       DØTARY(2)=2
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C       D0TARY(3)=3
C       D0TARY(4)=5
C       D0TARY(5)=21
C       D0TARY(6)=35
C       D0TARY(7)=46
C       D0TARY(8)=48
CCC
0149          IF(II.EQ.2)G0 T0 3000
0150          IF(II.EQ.3)G0 T0 52
0151            IF(NUMD0T.GT.0'G0 T0 80
0152            WRITE(6,1021)
0153    1021 F0RMAT(10X,'N0 D0TS F0UND')
0154          G0 T0 210
C
C   SET DISPLAY 0PTI0N ...
C
0155       80 C0NTINUE
0156          IDIS=2
0157            CALL 0UTPUT(27,12)
0158            WRITE(6,1020)
0159          WRI E(6,1007)
0160     1007 F0RMAT(10X,'SELECT DISPLAY 0PTI0N',/,12X,'1=WRITE 0VER',/,12X,'2=A
            1DD 0N -D0 N0T ERASE D0T',/,'$           3=ADD 0N -ERASE D0T    >')
0161          CALL 0UTPUT(7)
C
0162       85 READ(6,2000)W
0163          CALL FR0NT(W,74)
0164          IF(W(1).EQ.' ')G0 T0 100
0165          IF(W(1).EQ.'X')G0 T0 1009
0166          IF(W(1).EQ.'B')G0 T0 70
0167          IF(W(1).LT.'1')G0 T0 90
0168          IF(W(1).GT.'3')G0 T0 90
0169           IP=0
0170          CALL INTFF(IP,W,74,IDIS)
0171          G0 T0 100
C
0172       90 WRITE(6,1005)
0173          CALL 0UTPUT(7,7)
0174          G0 T0 85
C
0175      100 C0NTINUE
0176          MCX=1
0177          MCY=1
0178          IGRIDX=10
0179          IGRIDY=10
0180            IGRIDX=((MX*FL0AT(IGRIDX) + 0.5))
0181            IGRIDY=((MY*FL0AT(IGRIDY) + 0.5))
C
0182            MCX=((MX*FL0AT(MCX) + 0.5))
0183            MCY=((MY*FL0AT(MCY) + 0.5))
0184            IF(MX .LE. 1.0) MCX = 1
0185            IF(MY .LE. 1.0) MCY = 1
C
0186          IF(IDIS.NE.1)G0 T0 120
C
C   ERASE WINDØW FØR DISPLAY 0PTI0N SELECTED ...
```

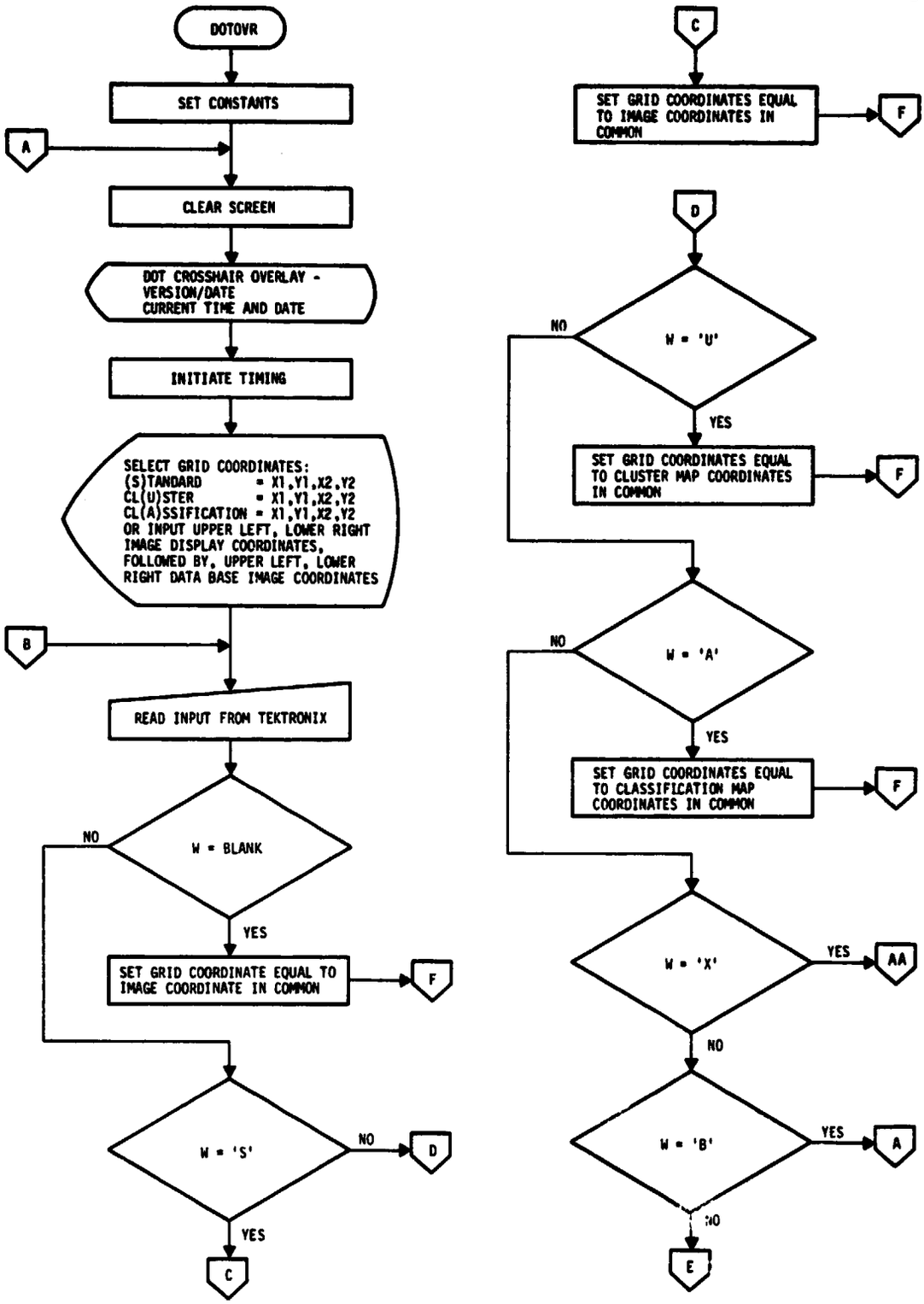
```
        C
0187            CALL BLKTHM(JX(1),JX(2),JX(3),JX(4),NT,0,1)
        C
0188          120 CONTINUE
        C
        C   COMPUTE CORNER POINT ...
0189            JX1=(MX*FLOAT(JX(5)-1))
0190            JX2=(MY*FLOAT(JX(6)-1))
        C
0191            JZ1=JX(1)-JX1
0192            JZ2=JX(2)-JX2
        C
        C       WRITE(5,1080)MX,MY,MCX,MCY,IGRIDX,IGRIDY,JZ1,JZ2,MXZ,MXY
0193    1080    FORMAT(6X,'MX,MY,...,MXZ,MXY',/,6X,2F6.2,14I6/)
        C       WRITE(5,1081)
0194    1081    FORMAT(///6X,' DEBUG PRINT'/)
        C
        C
        C   DRAW CROSS MARKS ON THEME ...
        C
0195            K=1
0196    8200      L=JZ2+IGRIDY-MXY-MCY
        C
0197            LL=2*MXY+MCY
0198            LM=LL/2
0199            LL=LL-LM
        C
        C
        C                       **** PROCESS EACH DOT ****
        C
0200            DO 219 I=0,NDOTS,19
0201            M=JZ1+IGRIDX-MCX
        C         WRITE(5,1041)L,M,LM,LL,MY,MX,JX,KWRT,NUMDOT
0202    1041      FORMAT(/6X,'PRINT AT 1041',4I8,2F6.2,12I6/)
        C     1   (DOTARY(III),III=1,NUMDOT)
        C         WRITE(5,9199) I,J,K,L,M,JX(2),JX(4)
0203    9199      FORMAT(6X,'I,J,K,L,M,JX(2),JX(4) 219',/,2X,7I6/)
0204     1040 FORMAT(16I6)
        C                                               ARE DOTS SELECTED
0205            IF(DOTARY(K).GT.I+19)GO TO 200
        C         WRITE(5,9190) L,MXY
0206    9190      FORMAT(/6X,'L,MXY 210',2I8/)
        C                                               INSIDE OF Y-COORDINATES
0207            IF(L+MXY.LT.JX(2))GO TO 200
0208            IF(L+MXY.GT.JX(4))GO TO 201
        C                                       READ LINES FROM THEME
0209            CALL IRT(NT,L,LL,NRA)
0210             CALL WAIT
        C
0211            CALL IRT(NT,L+1,LM,NRB)
0212            CALL WAIT
0213               KWRT=0
        C
0214            DO 188 J=1,19
0215            IF(DOTARY(K).NE.I+J)GO TO 188
        C
```

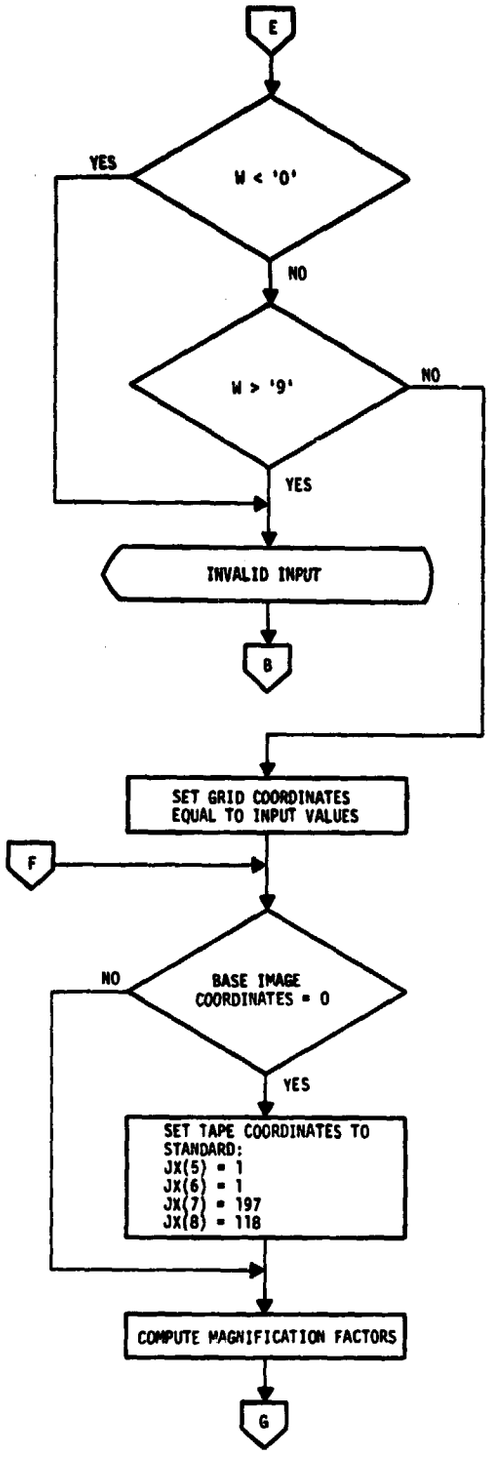
```
0216          IF(M.LT.JX(1))GØ TØ 180
0217          IF(M.GT.JX(3))GØ TØ 190
              C
              C                                   SET CRØSS MARKS IN BUFFER
              C
0218              LKA=1
0219              LKB=1
0220          KWRT=1
0221          LX=1
              C
              C       WRITE(5,1040)I,J,K,L,M
              C
0222          DØ 135 JI=1,MXY
0223          IF(LX.EQ.0)GØ TØ 130
0224          DØ 125 JJ=1,MCX
0225          LA=M+JJ-1
0226          CALL SETBIT(LA,NRA(LKA))
0227        125 CØNTINUE
0228          LKA=LKA+64
0229          LX=0
0230          GØ TØ 135
              C
0231        130 DØ 132 JJ=1,MCX
0232          LA=M+JJ-1
0233          CALL SETBIT(LA,NRB(LKB))
0234        132 CØNTINUE
0235          LKB=LKB+64
0236          LX=1
              C
0237        135 CØNTINUE
0238          DØ 160 JI=1,MCY
0239          IF(LX.EQ.0)GØ TØ 150
0240          DØ 140 JJ=1,MXZ
0241          LA=M-MXZ+JJ-1
0242          CALL SETBIT(LA,NRA(LKA))
0243        140 CØNTINUE
0244          IF(IDIS.NE.3)GØ TØ 146
0245          DØ 145 JJ=1,MCX
0246          LA=M+JJ-1
              C       MA=LA/8
              C       MB=LA-MA*8+1
              C       MC=MA+LKA
              C       NRA(MC)=IBIC(IMS(MB),NRA(MC))
0247          CALL SETBIT(LA,NRA(LKA))
0248        145 CØNTINUE
0249        146 DØ 148 JJ=1,MXZ
0250          LA=M+MCX+JJ-1
0251          CALL SETBIT(LA,NRA(LKA))
0252        148 CØNTINUE
0253          LKA=LKA+64
0254          LX=0
0255          GØ TØ 160
              C
0256        150 DØ 154 JJ=1,MXZ
0257          LA=M-MXZ+JJ-1
0258          CALL SETBIT(LA,NRB(LKB))
```

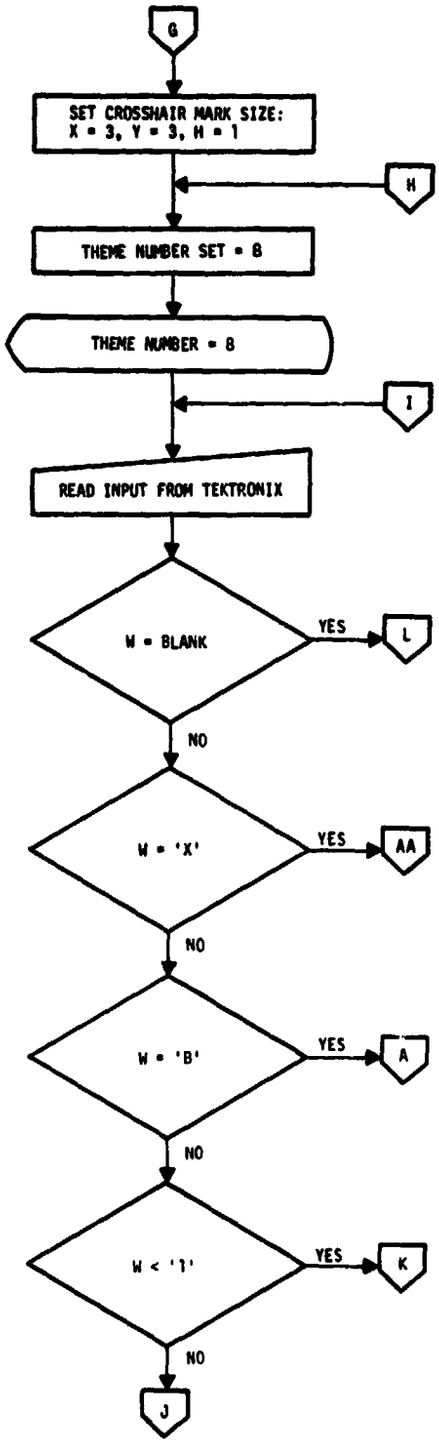
```
0259      154 CØNTINUE
          C
0260            IF(IDIS.NE.3)GØ TØ 157
0261            DØ 156 JJ=1,MCX
0262            LA=M+JJ-1
          C      MA=LA/8
          C      MB=LA-MA*8+1
          C      MC=MA+LK4
          C      NRB(MC)=IBIC(IMS(MB),NRB(MC))
0263            CALL SETBIT(LA,NRB(LKB))
0264      156 CØNTINUE
          C
0265      157   DØ 158 JJ=1,MX2
0266            LA=M+MCX+JJ-1
0267            CALL SETBIT(LA,NRB(LKB))
0268      158 CØNTINUE
          C
0269            LKB=LKB+64
0270            LX=1
0271      160 CØNTINUE
          C
          C
0272            DØ 175 JI=1,MXY
0273            IF(LX.EQ.0)GØ TØ 165
0274            DØ 162 JJ=1,MCX
0275            LA=M+JJ-1
0276            CALL SETBIT(LA,NRA(LKA))
0277      162 CØNTINUE
0278            LKA=LKA+64
0279            LX=0
0280            GØ TØ 175
          C
0281      165   DØ 168 JJ=1,MCX
0282            LA=M+JJ-1
0283            CALL SETBIT(LA,NRB(LKB))
0284      168 CØNTINUE
0285            LKB=LKB+64
0286            LX=1
          C
0287      175 CØNTINUE
0288            K=K+1
0289            IF(K.GT.NUMDØT)GØ TØ 190
0290      180     M=M+IGRINX
0291      185 CØNTINUE
          C
0292      190 CØNTINUE
          C       WRITE(5,155) KWRT
0293      155     FØRMAT(6X,'KWRT AFTER 190',I8/)
          C       WRITE(5,155) KWRT
0294            IF(KWRT.EQ.0)GØ TØ 200
          C
          C              ... WRITE CRØSSHAIR MARKS ØN THEME ...
          C
0295            CALL IWT(NT,L,LL,NRA)
0296            CALL WAIT
          C
```

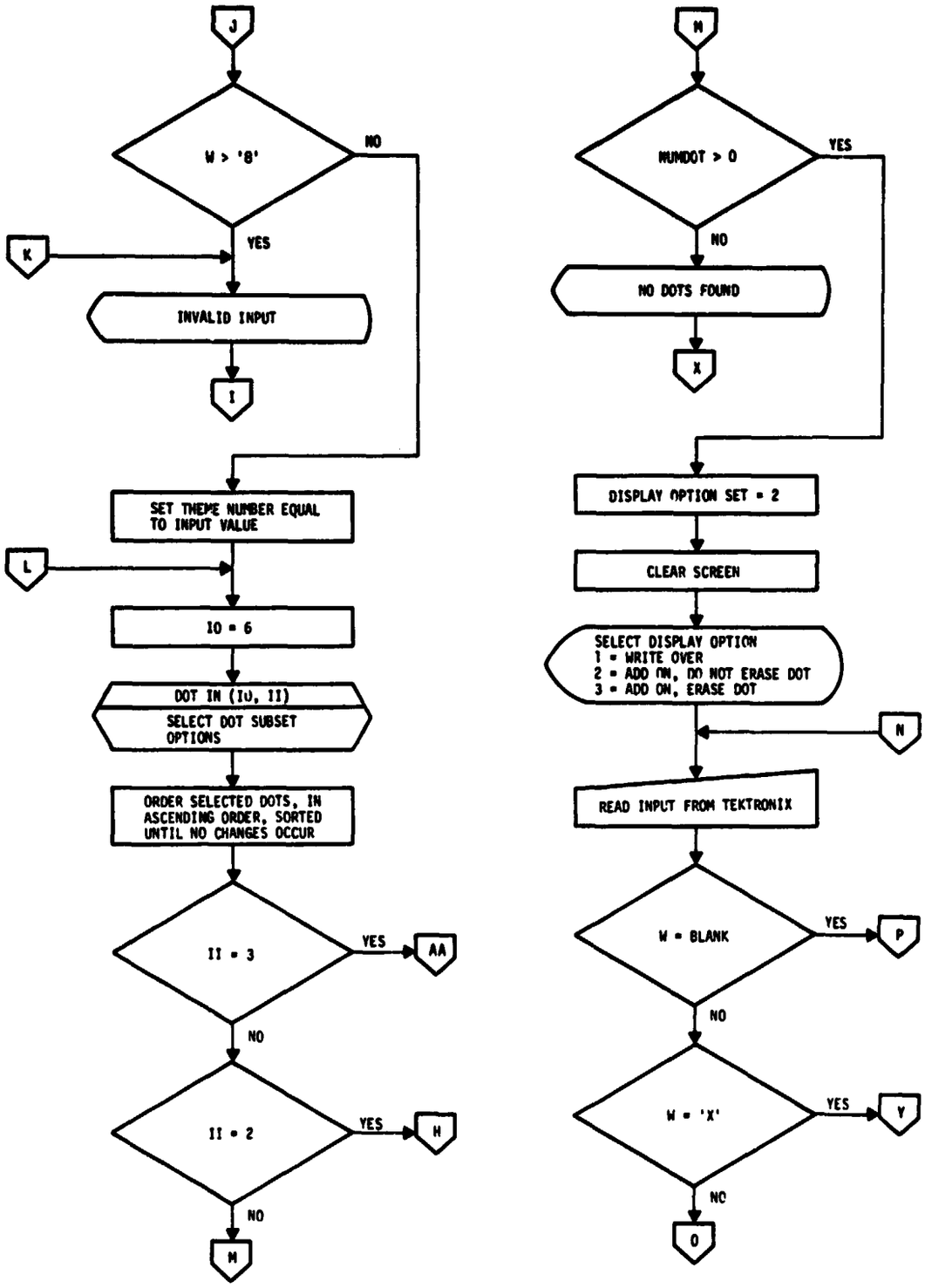
```
0297            CALL IWT(NT,L+1,LM,NRB)
0298            CALL WAIT
0299             IF(K .GT. NUMDØT) GØ TØ 210
0300             GØ TØ 8200
        C
        C
0301    200     L=L+IGRIDY
0302    219     CØNTINUE
0303    201     K=K+1
0304             IF(K .GT. NUMDØT) GØ TØ 210
0305             GØ TØ 8200
        C
0306      210 CØNTINUE
        C       WRITE(5,1899) I,J,K,L,M
0307    1899    FØRMAT(6X,'I,J,K,L,M  AFTER 210',/,6X,5I10/)
        C       WRITE(5,1042)
0308    1042    FØRMAT(///6X,'FINISH'/)
        C                    GET ELAPSE TIME ...
0309            T1=SECØNDS(TI)
0310            WRITE(6,1010)T1
0311     1010 FØRMAT(10X,'=ELAPSE TIME = ',F9.3,' SECØNDS')
        C
0312    1009    WRITE(6,1011)
0313     1011 FØRMAT('$      (R)ECYCLE ØR E(X)IT >')
0314            CALL ØUTPUT(7)
0315      216 READ(6,2000)W
0316            CALL FRØNT(W,74)
0317            IF(W(1).EQ.'X')GØ TØ 3000
0318            IF(W(1).EQ.'R')GØ TØ 5
0319            IF(W(1).EQ.' ')GØ TØ 3000
0320            WRITE(6,1005)
0321            GØ TØ 216
        C
        C                    -- DØTØVR -- TERMINATIØN ...
        C
0322     3000 CØNTINUE
        C       CALL DMP(0)
0323            CALL SETEF(50)
0324               CALL DETACH
0325            WRITE(6,1012)
0326     1012 FØRMAT(10X,'DØT CRØSSHAIR ØVERLAY CØMPLETED.... ... .. . . ')
        C
        CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
        C                                                               C
        C                                                               C
        CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
0327            END
```

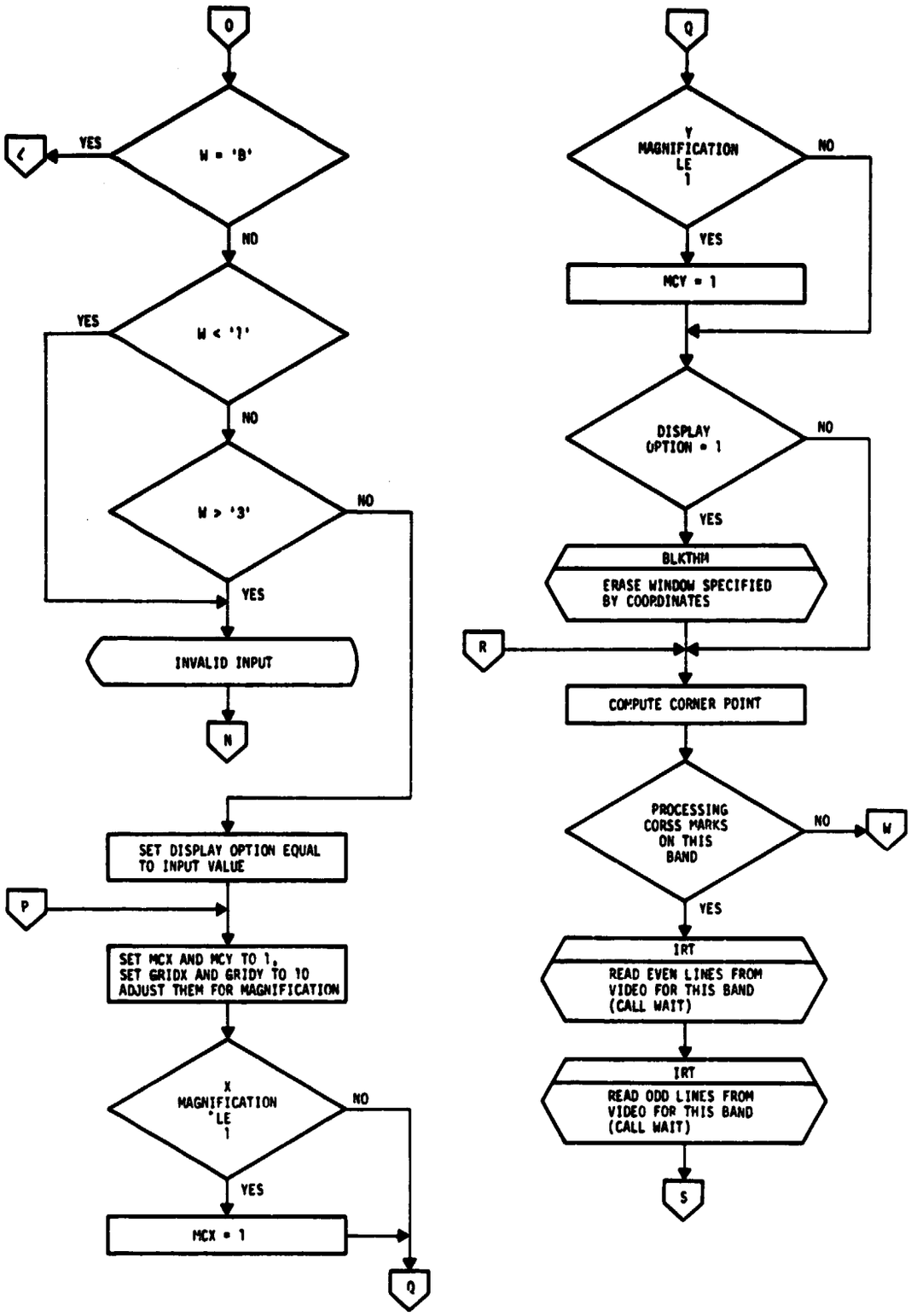
ORIGINAL PAGE IS
OF POOR QUALITY

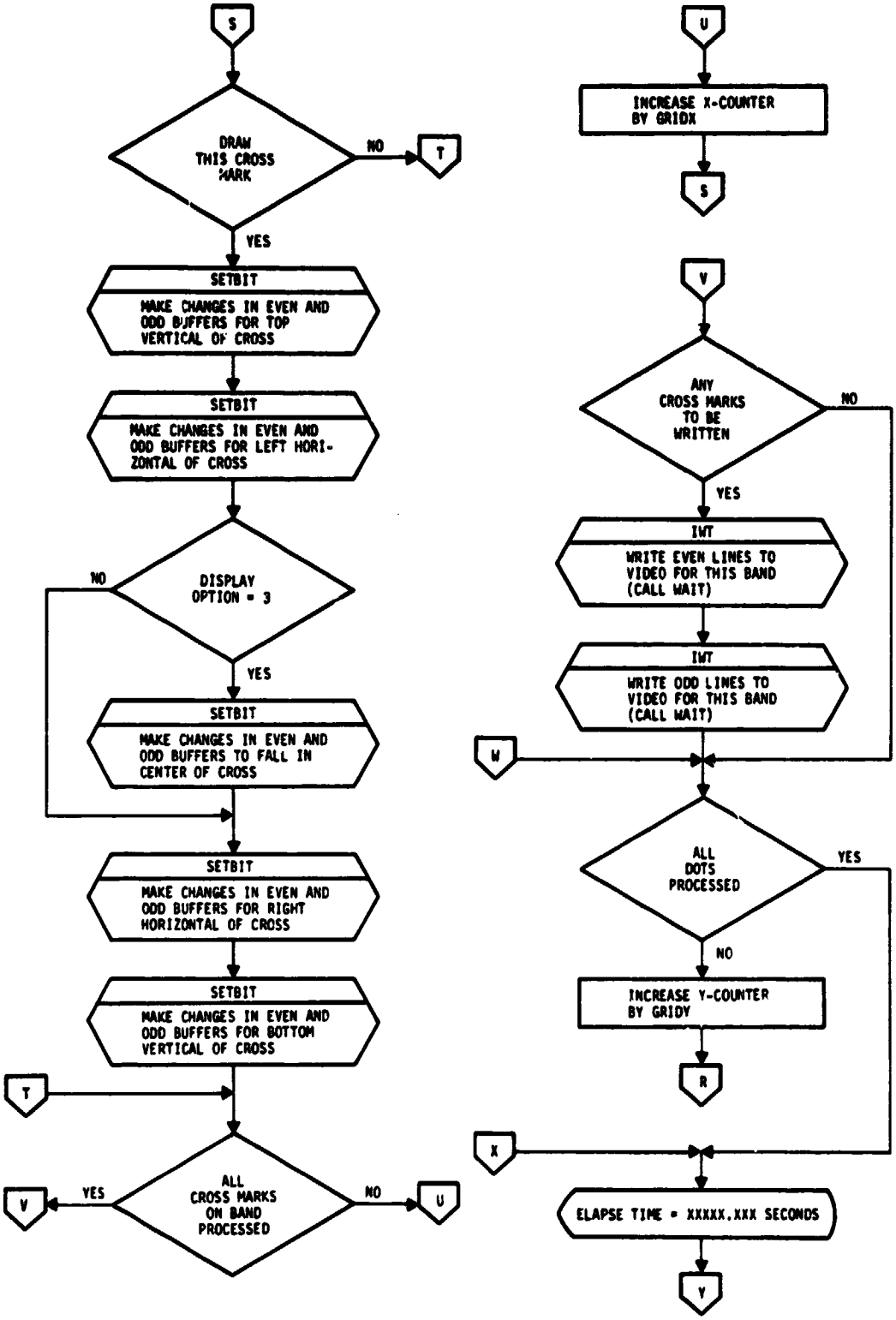


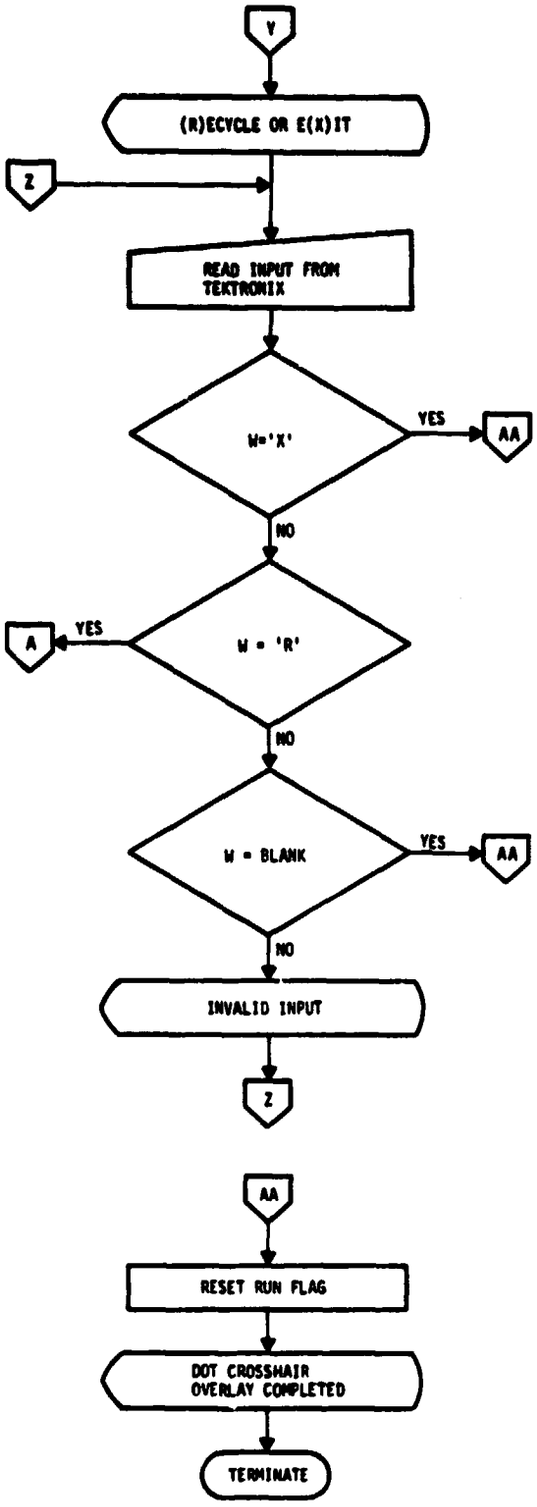

E
W < '0'
YES
NO
W > '9'
NO
YES
INVALID INPUT
B
SET GRID COORDINATES EQUAL TO INPUT VALUES
F
BASE IMAGE COORDINATES = 0
NO
YES
SET TAPE COORDINATES TO STANDARD:
JX(5) = 1
JX(6) = 1
JX(7) = 197
JX(8) = 118
COMPUTE MAGNIFICATION FACTORS
G


G
SET CROSSHAIR MARK SIZE:
X = 3, Y = 3, M = 1
H
THEME NUMBER SET = 8
THEME NUMBER = 8
I
READ INPUT FROM TEKTRONIX
W = BLANK
YES
L
NO
W = 'X'
YES
AA
NO
W = 'B'
YES
A
NO
W < '1'
YES
K
NO
J


J
W > '8'
NO
YES
K
INVALID INPUT
I
SET THEME NUMBER EQUAL TO INPUT VALUE
L
IO = 6
DOT IN (IO, II)
SELECT DOT SUBSET OPTIONS
ORDER SELECTED DOTS, IN ASCENDING ORDER, SORTED UNTIL NO CHANGES OCCUR
II = 3
YES
AA
NO
II = 2
YES
H
NO
M
M
NUMDOT > 0
YES
NO
NO DOTS FOUND
X
DISPLAY OPTION SET = 2
CLEAR SCREEN
SELECT DISPLAY OPTION
1 = WRITE OVER
2 = ADD ON, DO NOT ERASE DOT
3 = ADD ON, ERASE DOT
N
READ INPUT FROM TEKTRONIX
W = BLANK
YES
P
NO
W = 'X'
YES
Y
NO
O


O
W = 'B'
YES
NO
W < '1'
YES
NO
W > '3'
NO
YES
INVALID INPUT
N
SET DISPLAY OPTION EQUAL TO INPUT VALUE
P
SET MCX AND MCY TO 1, SET GRIDX AND GRIDY TO 10 ADJUST THEM FOR MAGNIFICATION
X MAGNIFICATION LE 1
NO
YES
MCX = 1
Q
Q
Y MAGNIFICATION LE 1
NO
YES
MCY = 1
DISPLAY OPTION = 1
NO
YES
BLKTHM
ERASE WINDOW SPECIFIED BY COORDINATES
R
COMPUTE CORNER POINT
PROCESSING CORSS MARKS ON THIS BAND
NO
W
YES
IRT
READ EVEN LINES FROM VIDEO FOR THIS BAND (CALL WAIT)
IRT
READ ODD LINES FROM VIDEO FOR THIS BAND (CALL WAIT)
S


S
DRAW THIS CROSS MARK
NO
T
YES
SETBIT
MAKE CHANGES IN EVEN AND ODD BUFFERS FOR TOP VERTICAL OF CROSS
SETBIT
MAKE CHANGES IN EVEN AND ODD BUFFERS FOR LEFT HORIZONTAL OF CROSS
DISPLAY OPTION = 3
NO
YES
SETBIT
MAKE CHANGES IN EVEN AND ODD BUFFERS TO FALL IN CENTER OF CROSS
SETBIT
MAKE CHANGES IN EVEN AND ODD BUFFERS FOR RIGHT HORIZONTAL OF CROSS
SETBIT
MAKE CHANGES IN EVEN AND ODD BUFFERS FOR BOTTOM VERTICAL OF CROSS
T
ALL CROSS MARKS ON BAND PROCESSED
YES
V
NO
U
U
INCREASE X-COUNTER BY GRIDX
S
V
ANY CROSS MARKS TO BE WRITTEN
NO
YES
IWT
WRITE EVEN LINES TO VIDEO FOR THIS BAND (CALL WAIT)
IWT
WRITE ODD LINES TO VIDEO FOR THIS BAND (CALL WAIT)
W
ALL DOTS PROCESSED
YES
NO
INCREASE Y-COUNTER BY GRIDY
R
X
ELAPSE TIME = XXXXX.XXX SECONDS
Y


Y
(R)ECYCLE OR E(X)IT
Z
READ INPUT FROM TEKTRONIX
W='X'
YES
AA
NO
W = 'R'
YES
A
NO
W = BLANK
YES
AA
NO
INVALID INPUT
Z
AA
RESET RUN FLAG
DOT CROSSHAIR OVERLAY COMPLETED
TERMINATE

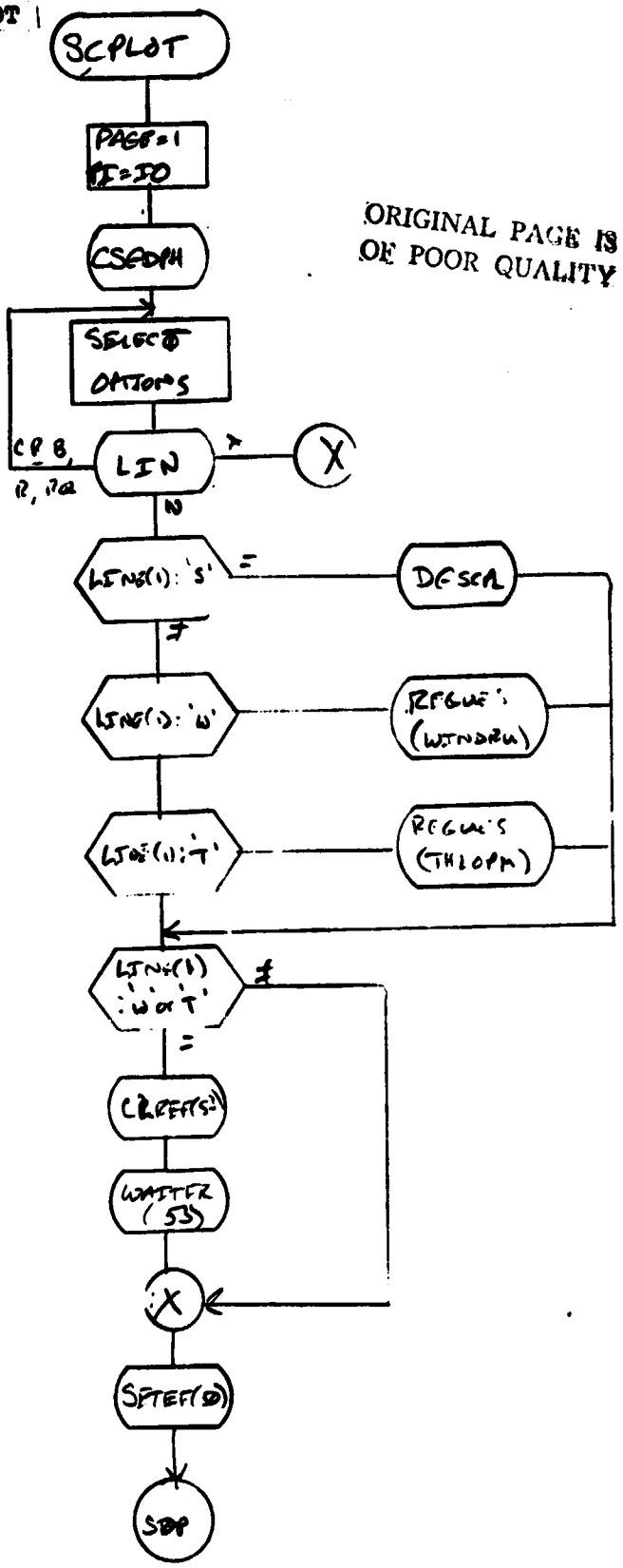
```
HFORTRAN IV-PLUS V02-04          13158144    06-SEP-77          PAGE 1
SCPLOT.FTN      /TRIMLOCKS/WR
0001            INCLUDE 'TOP.INC'
      *  C!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!C
      *  C                                                                      C
      *  C                                                                      C
      *  C                                                                      C
      *  C                                                                      C
         C                SCATTER PLOT DRIVER PROGRAM.                          C
         C                T. KELLY/LEC/4/77                                     C
         C                SCPLOT.FTN                                            C
0002            INCLUDE 'BOT.INC'
      *  C                                                                      C
      *  C                                                                      C
      *  C                                                                      C
      *  C                                                                      C
      *  C!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!C
0003            PROGRAM  SCPLOT
0004            INCLUDE 'HUGGYBEAR.CMN'
0005 *          INCLUDE '[300,3]CAMSCOMOT.INC'
0006 *          INCLUDE 'SYS[300,3]CAMSPARAM.INC'
0007 *          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=90
      *         1,MAXV=11,NDOTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *         2NOSPHD=6,NODTWD=10
0008 *          EQUIVALENCE (C1,ACDATE),(C2,ISEG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0009 *          INTEGER C1(469),C2(256),C3(71),C4(348),C5(629)
      *  C*
0010 *          INTEGER ACDATE,SUBCAT,SUBPOP,CATNAM,CATTH
0011 *          BYTE CHNVEC,NOCHAN,NOSUB,DOTCAT,DOTCLU
0012 *          COMMON/COM1/ACDATE(2,MAXACC),CHNVEC(MAXCHN,MAXACC),NOCHAN,NOSUB,
      *         1SUBCAT(MAXSUB),SUBPOP(MAXSUB),CATNAM(MAXCAT),CATTH(MAXCAT),NODO,
      *         2NODU,CATH,DOTCAT(NODTS),DOTCLU(NODTS)
      *  C*
0013 *          INTEGER ADATES,SUNAZ,ANALST,FLDDAY,DOTDAY,PDATE1,TDATE1
0014 *          INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RANDOM,GRID
0015 *          BYTE DELFLG,NOACQ,SOILGR,SUNEL,NSTART,NTYPE1,ALP,ALP0
0016 *          BYTE POTCT,PCTCT0,VAR,VAR0,DLABEL,TYPE
0017 *          COMMON/COM2/ISEG,DELFLG,NOACQ,ADATES(2,MAXACD),SOILGR(MAXACD),
      *         1SUNEL(MAXACD),SUNAZ(MAXACD),IMDATE(2),ANALST(5),FLDDAY(2),
      *         2DOTDAY(2),NSTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      *         3PDATE3(2),TDATE3(2),NOCAT,CATNAM(MAXCAT),ALP(MAXCAT),ALP0,
      *         4          PCTCT(MAXCAT),PCTCT0,VAR(MAXCAT),VAR0
      *  C*
0018 *          INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      *         1UFLAG4
0019 *          INTEGER PFLAG,DSKCNT
0020 *          COMMON/COM3/PFLAG,DSKCNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
      *         1,UFLAG2,UFLAG3,UFLAG4,NEWLAB(MAXSUB)
      *  C*
0021 *          INTEGER TX1,TY1,TX2,TY2,ACDISP,G,B,DYWIND,DOTANY,GMIN,GMAX,FUL
0022 *          INTEGER SPWIND,CLAWND,CLUWND
0023 *          COMMON/COM4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACDISP(2),I11(4),G(4),
      *         1B(4),DTWIND(5,NODTWD),SPWIND(5,NOSPHD),IMWIND(4),NUMDOT,
      *         2DOTANY(NDOTS),GMIN,GMAX,FUL(2,7),CLAWND(8),CLUWND(8)
0024 *          COMMON/COM5/DISKID,RANDOM(NDOTS),GRID(NDOTS),DLABEL(NDOTS),
      *         1TYPE(NDOTS),RECLOC
0025 *            BYTE ACDAC,ACDAI,ACDAJ,ACDAD,BUF,DB,LINE,VAL
0026 *            BYTE PASS,PCH,THR

                                  12-1

                                  ORIGINAL PAGE IS
                                  OF POOR QUALITY
```

```
0027 *          INTEGER CIF,DFLAG,DATE,HMIN,HMAX,I,J,K,LIST,ND,TM,VMIN
0028 *          INTEGER VMAX,XA,XI,YA,YI,TI
0029 *          INTEGER PTR,RECL,WN
0030 *          REAL BXY,SXY,XYI,XYA
0031 *          COMMON /HUGGY/ ACDAC,ACDAI,ACUAJ,ACDAD,BUF(512,4),CIF,
     *         1 DFLAG,DATE(2),DB(10,ND0TS+1),HMIN,HMAX,I,J,K,LINE(74),
     *         2 LIST(ND0TS),ND,VAL,PASS,PXY(2),PCH(2),PTR(ND0TS+1),RECL,
     *         3 WN,SXY(2),TM,TM5(64),VMIN,VMAX,XA,XI,XYA(2),XYI(2),YA,YI,TI
0032 *          INTEGER DBW(5,ND0TS+1)
0033 *          EQUIVALENCE(DB(1,1),DBW(1,1))
0034 *          BYTE INRAY(36*MAXACC),D0TS(8,MAXACD),D0TP(8)
0035 *          INTEGER CLUSW(18,MAXACC),D0TSW(4,MAXACD)
0036 *          REAL CLUS(9,MAXACC)
0037 *          EQUIVALENCE (BUF(1,1),INRAY(1))
0038 *          EQUIVALENCE(INRAY(1),D0TP(1))
0039 *          EQUIVALENCE (D0TP(5),D0TS(1,1))
0040 *          EQUIVALENCE (D0TP(5),D0TSW(1,1))
0041 *          EQUIVALENCE (INRAY(1),CLUS(1,1))
0042 *          EQUIVALENCE (CLUS(1,1),CLUSW(1,1))
0043             REAL THM,WIN
0044             BYTE ILINE(74)
0045             INTEGER PAGE
0046             DATA THM,WIN/6RTHLOPM,6RWINDPW/
0047             DATA I0/0/
0048             DATA LP/6/
0049             PAGE = 1
0050             TI = 12
0051             CALL CSGDPH(12,PAGE,'SCPLOT 1.0 4/7/77',17)
0052     1       WRITE(10,1000)
0053     1000    FORMAT(' SELECT OPTIONS: (S)CATTER PLOT, (W)INDOW',
                1 ' ERASE, OR',/,'(T)HEME LOGICAL OPERATIONS. >')
0054             CALL LIN(10,ILINE,TI,DFLAG)
0055             GO TO (2,1,1,1,1,3)DFLAG
0056     3       CONTINUE
0057             IF(ILINE(1).EQ.'S')CALL SCPLO(.0,LP)
0058             IF(ILINE(1).EQ.'W') CALL REQUES(WIN)
0059             IF(ILINE(1).EQ.'T') CALL REQUES(THM)
0060             IF(ILINE(1).EQ.'E'.OR.ILINE(1).EQ.'T')CALL CLREF(53)
0061             IF(ILINE(1).EQ.'E'.OR.ILINE(1).EQ.'T')CALL WAITFR(53)
0062     2       CONTINUE
0063             INCLUDE '[300,3]CAMSAVE.INC'
0064 *          OPEN(UNIT=1,NAME='[300,1]GLOBAL.TMP;1',FORM='UNFORMATTED',
     *       1   TYPE='UNKNOWN',ERR=9900)
0065 *            WRITE(1,C1
0066 *            WRITE(1)C2
0067 *            WRITE(1)C3
0068 *            WRITE(1)C4
0069 *            WRITE(1)C5
0070 *            CLOSE(UNIT=1)
0071 *            GO TO 9991
0072 *  9900      TYPE 9990
0073 *  9990      FORMAT(1X,'OPEN FAILURE ON [300,1]GLOBAL.TMP--NO RESTART')
0074 *  9991      CONTINUE
0075             CALL SETEF(50)
0076             STOP
0077             END
```

12. DOT GROUP SCATTER PLOT SCPLOT



ORIGINAL PAGE IS
OF POOR QUALITY

12.1 SUBROUTINE DGSCPL

```
MFORTRAN IV-PLUS V02-04          14:04:16    31-AUG-77          PAGE 1
DGSCPL.FTN      /TR:BL@CKS/WR
0001          SUBR@UTINE DGSCPL(I@,LP)
0002          INCLUDE 'T@P.INC'
      *  CICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCIC
      *  C                                                                     C
      *  C                                                                     C
      *  C                                                                     C
      *  C                                                                     C
      *  C                                                                     C
      *  C                D@T/CLUSTER:SCATTER:PLOT                             C
      *  C                T.JKELL/LEC/4/77                                     C
      *  CDGSCPL.FTN                                                          :::
0003          INCLUDE 'B@T.INC'
      *  C                                                                     C
      *  C                                                                     C
      *  C                                                                     C
      *  C                                                                     C
      *  CICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCICCIC
0004          INCLUDE:BUGGYBEAR.C@M'
0005  *         INCLUDE '[300,3]CAMSC@M@N.INC'
0006  *       INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0007  *       PARAMETER MAXCAT=6@,MAXSUB=6@,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      *      1,MAXV=11,ND@TS=209,DLSKIP=10,USSKIP=10,MAXACD=6,MAXACC=4,
      *      2N@SPWD=6,N@DTWD=10
0008  *       EQUIVALENCE (C1,ACDATE),(C2,ISEG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0009  *       INTEGER C1(469),C2(256),C3(71),C4(348),C5(629)
      *  C*
0010  *       INTEGER ACDATE,SUBCAT,SUBP@P,CATKNT,CATTH
0011  *       BYTE CHNVEC,N@CHAN,N@SUB,D@TCAT,D@TCLU
0012  *       C@MM@N/C@M1/ACDATE(2,MAXACC),CHNVEC(MAXCHN,MAXACC),N@CHAN,N@SUB,
      *      1SUBCAT(MAXSUB),SUBP@P(MAXSUB),CATKNT(MAXCAT),CATTH(MAXCAT),N@D@,
      *      2N@DU,N@TH,D@TCAT(ND@TS),D@TCLU(ND@TS)
      *  C*
0013  *       INTEGER ADATES,SUNAZ,ANALST,FLDDAY,D@TDAY,PDATE1,TDATE1
0014  *       INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RAND@M,GRID
0015  *       BYTE DELFLG,N@ACD,S@ILGR,SUNEL,NSTART,NTYPE1,ALP,ALP@
0016  *       BYTE PCTCT,PCTCT@,VAP,VAR@,DLABEL,TYPE
0017  *       C@MM@N/C@M2/ISEG,DELFLG,N@ACD,ADATES(2,MAXACD),S@ILGR(MAXACD),
      *      1SUNEL(MAXACD),SUNAZ(MAXACD),IMDATE(2),ANALST(5),FLDDAY(2),
      *      2D@TDAY(2),NSTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      *      3PDATE3(2),TDATE3(2),N@CAT,CATNAM(MAXCAT),ALP(MAXCAT),ALP@,
      *      4          PCTCT(MAXCAT),PCTCT@,VAR(MAXCAT),VAR@
      *  C*
0018  *       INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      *      1UFLAG4
0019  *       INTEGER PFLAG,DSKMNT
0020  *       C@MM@N/C@M3/PFLAG,DSKMNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
      *      1,UFLAG2,UFLAG3,UFLAG4,NEWLAB(MAXSUB)
      *  C*
0021  *       INTEGER TX1,TY1,TX2,TY2,ACDISP,G,B,DTWIND,D@TARY,GMIN,GMAX,FUL
0022  *       INTEGER SPWIND,CLAWND,CLUWND
0023  *       C@MM@N/C@M4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACDISP(2),I11(4),G(4),
      *      1B(4),DTWIND(5,N@DTWD),SPWIND(5,N@SPWD),IMWIND(4),NUMD@T,
      *      2D@TARY(ND@TS),GMIN,GMAX,FUL(2,7),CLAWND(8),CLUWND(8)
0024  *       C@MM@N/C@M5/DISKID,RAND@M(ND@TS),GRID(ND@TS),DLABEL(ND@TS),
      *      1TYPE(ND@TS),RECL@C
0025  *         BYTE ACDAC,ACDAI,ACDAJ,ACDAD,BUF,DR,LINE,VAL
0026  *         BYTE PASS,PCH,THR
```

```
DGSCPL.FTN      /TR:BLØCKS/WR
0027 *          INTEGER CIF,DFLAG,DATE,HMIN,HMAX,I,J,K,LIST,ND,TH,VMIN
0028 *          INTEGER VMAX,XA,XI,YA,YI,TI
0029 *          INTEGER PTR,RECL,WN
0030 *          REAL ØXY,SXY,XYI,XYA
0031 *          CØMMØN /HUGGY/ ACDAC,ACDAI,ACDAJ,ACDAD,BUF(512,4),CIF,
     *         1 DFLAG,DATE(2),DB(10,NDØTS+1),HMIN,HMAX,I,J,K,LINE(74),
     *         2 LIST(NDØTS),ND,VAL,PASS,ØXY(2),PCH(2),PTR(NDØTS+1),RECL,
     *         3 WN,SXY(2),TH,THB(64),VMIN,VMAX,XA,XI,XYA(2),XYI(2),YA,YI,TI
0032 *          INTEGER DBW(5,NDØTS+1)
0033 *          EQUIVALENCE(DB(1,1),DBW(1,1))
0034 *          BYTE INRAY(36*MAXACC),DØTS(8,MAXACD),DØTP(6)
0035 *          INTEGER CLUSW(18,MAXACC),DØTSW(4,MAXACD)
0036 *          REAL CLUS(9,MAXACC)
0037 *          EQUIVALENCE (BUF(1,1),INRAY(1))
0038 *          EQUIVALENCE(INRAY(1),DØTP(1))
0039 *          EQUIVALENCE (DØTP(5),DØTS(1,1))
0040 *          EQUIVALENCE (DØTP(5),DØTSW(1,1))
0041 *          EQUIVALENCE (INRAY(1),CLUS(1,1))
0042 *          EQUIVALENCE (CLUS(1,1),CLUSW(1,1))
0043            INTEGER MIN(2)
0044            BYTE CHN(7)
0045            INTEGER WIN(5,NØSPWD)
0046            EQUIVALENCE (SPWIND(1,1),WIN(1,1))
0047          BYTE AA(2),ALL(3),SCA
0048            INTEGER FFUNC
0049            INTEGER VC,HC
0050            DATA CHN/'1','2','3','4','G','B','N'/
0051          CIF = 1
0052            RECL = 0
0053            CALL ELAPSE(1)
0054            TH = 1
0055          AA(1)='D'
0056          AA(2)='C'
0057          ALL(1)='A'
0058          ALL(2)='L'
0059          ALL(3)='L'
0060          SCA = 'G'
0061            HC = 'B'
0062            VC = 'N'
0063            IPA = 1
0064    1     PASS = 0
0065            CALL CSGØPH(I0,IPA,'DGSCPL 1.0 4/7/77',17)
0066            CALL SETWIN(I0)
0067          GØ TØ (100,1,100,1,2,2)DFLAG
0068    2     IF(WN.GT.0) GØ TØ 3
0069          WRITE(I0,1000)
0070          TI = I0
0071    1000  FØRMAT(' NØ WINDØW AVAILABLE')
0072          GØ TØ 100
0073    3       CALL CSGØPH(I0,IPA,'DGSCPL 1.0 4/7/77',17)
0074            IF(TI.EQ.I0)WRITE(I0,1001)AA(CIF)
0075    1001  FØRMAT('$SELECT ØPTIØNS FØR SCATTER PLØTS: (D)ØT GRØUP ØR (C)LUSTE
     1R MEANS. ',A1,' >')
0076          CALL LIN(I0,LINE,TI,DFLAG)
0077          GØ TØ (100,1,100,3,4,5)DFLAG
0078    5     A = 0
```

```
0079            IF(LINE(1),EQ,AA(1))A = 1
0080            IF(LINE(1),EQ,AA(2))A = 2
0081            IF(A,NE,0) GO TO 3333
0082              TI = 10
0083              WRITE(10,1018)
0084    3333      CIF = A
0085    4       IF(PASS,EQ,0) GO TO 6
0086            IF(CIF,NE,1) GO TO 7
0087              IF(TI,EQ,10)WRITE(10,1002)((ADATES(J,I),J=1,2),I=1,MAXACD)
              1 ,DATE,HC,VC
0088    1002    FORMAT(' ACQUISITIONS AVAILABLE FOR DOT GROUT SCATTER PLOT:'/
              1 ' ',12X,'ACQ. DATES  CHANNEL'/(MAXACD)(' ',14X,213,8X,'ALL'/),
              2 ' FOR THE CURRENT WINDOW IN USE :'/'    ACQUISITION DATE = '213/
              3 '    CHANNELS = ',A1,' , ',A1/')
0089            DO 8 I = 1,MAXACD
0090            IF(DATE(1),NE,ADATES(1,I)) GO TO 8
0091            IF(DATE(2),EQ,ADATES(2,I)) GO TO 9
0092    8       CONTINUE
0093    15      WRITE(10,1003)
0094    1003    FORMAT(' CAN NOT PROCEED')
0095              TI = 10
0096            GO TO 3
0097    9       ACDAD = I
0098            ACDAI = I
0099    10      IF(TI,EQ,10)WRITE(10,1004)
0100    1004    FORMAT('PROCEED (Y)ES/(N)? >')
0101            CALL LIN(10,LINE,TI,DFLAG)
0102            GO TO (100,3,100,10,10,11)DFLAG
0103    11      IF(LINE(1),EQ,'Y') GO TO 12
0104            IF(LINE(1),EQ,'N') GO TO 3
0105            GO TO 10
0106    7         IF(TI,EQ,10)WRITE(10,1005)((ACDATE(J,I),J=1,2),(CHNVEC(J,I),J=1,4),
                  1 I=1,MAXACC),DATE,HC,VC
0107    1005    FORMAT(' ACQUISITIONS AVAILABLE FOR CLUSTER MEAN SCATTER PLOT:'/'
              1',12X,'ACQ. DATE   CHANNEL'/(MAXACC)(' ',14X,213,5X,4(I1,1X)/)/
                4 ' FOR THE CURRENT WINDOW IN USE:'/
              2 '    ACQUISITION DATE = ',213/'    CHANNELS = ',A1,' , ',
              3A1/)
0108            DO 13 I = 1,MAXACC
0109            IF(DATE(1),NE,ACDATE(1,I)) GO TO 13
0110            IF(DATE(2),EQ,ACDATE(2,I)) GO TO 14
0111    13      CONTINUE
0112            GO TO 15
0113    14      ACDAI = I
0114            ACDAC = I
0115            CALL VALCK
0116            IF(VAL,EQ,0) GO TO 15
0117            GO TO 10
0118    6       DO 16 I = 1,MAXACD
0119              IF(CIF,EQ,2) GO TO 162
0120            IF(ACDISP(1),NE,ADATES(1,I))GO TO 16
0121            IF(ACDISP(2),NE,ADATES(2,I)) GO TO 16
0122            ACDAJ = I
0123            ACDAD = I
0124            ACDAI = I
0125              GO TO 16
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
0126    162     IF(ACDISP(1).NE.ACDATE(1,I)) GØ TØ 16
0127            IF(ACDISP(2).NE.ACDATE(2,I)) GØ TØ 16
0128            ACDAJ = I
0129            ACDAC = I
0130            ACDAI = I
0131    16      CØNTINUE
0132    17      IF(TI.EQ.I2) WRITE(I2,1006)
0133    1006    FØRMAT(' SELECT ACQUISITIØN DATA FRØM THE FØLLØWING DATES')
0134            IF(CIF.EQ.2) GØ TØ 18
0135            IF(TI.EQ.I2)WRITE(I2,1007)
0136    1007    FØRMAT(' FØR DØT GRØUP SCATTER PLØT')
0137            ACDAI = ACDAC
0138            N = MAXACD
0139            IF(TI.EQ.I2)WRITE(I2,1008)(I,(ADATES(J,I),J=1,2),I=1,MAXACD),ACDAI
0140    1008    FØRMAT(<N-1>(' (',I1,') ',2I3/),('S(',I1,') ',2I3,10X,I1,10X,'>'))
0141            GØ TØ 19
0142    18      IF(TI.EQ.I2)WRITE(I2,1009)
0143    1009    FØRMAT(' FØR CLUSTER MEANS SCATTER PLØT')
0144            ACDAI = ACDAC
0145            N = MAXACC
0146            IF(TI.EQ.I2)WRITE(I2,1008)(I,(ACDATE(J,I),J=1,2),I=1,MAXACC),ACDAI
0147    19      CALL LIN(I2,LINE,T',DFLAG)
0148            GØ TØ (100,3,100,17,21,20)DFLAG
0149    20      I = 0
0150            CALL INTFF(I,LINE,74,IAC)
0151              ACDAI = IAC
0152    21      IF(0.LT.ACDAI.AND.ACDAI.LE.N) GØ TØ 117
0153            TI = I2
0154            WRITE(I2,1018)
0155            GØ TØ 17
0156    117     IF(3*CIF-6.LT.0) GØ TØ 22
0157            ACDAC = ACDAI
0158            DATE(1)=ACDATE(1,ACDAC)
0159            DATE(2)=ACDATE(2,ACDAC)
0160            GØ TØ 23
0161    22      ACDAD = ACDAI
0162            DATE(1)=ADATES(1,ACDAD)
0163            DATE(2)=ADATES(2,ACDAD)
0164    23      IF(TI.EQ.I2)WRITE(I2,1010)HC,VC
0165    1010    FØRMAT(' SELECT THE CØØRDINATE SYSTEM!'/' CHANNELS 1 - 4, (G)REENN
              1ESS, (B)RIGHTNESS, ØR GREEN (N)UMBER,'/'$ ',A1,',',A1,' >')
0166            CALL LIN(I2,LINE,TI,DFLAG)
0167            GØ TØ (100,17,100,23,24,25)DFLAG
0168    24      LINE(1)=HC
0169            LINE(3)=VC
0170            IF(TI.EQ.I2)WRITE(I2,1011)HC,VC
0171    1011    FØRMAT(' SELECTED CHANNELS = ',A1,',',A1)
0172    25      DØ 26 I=1,3,2
0173            DØ 27 J=1,7
0174    27      IF(LINE(I).EQ.CHN(J))GØ TØ 28
0175            TI = I2
0176            WRITE(I2,1018)
0177            GØ TØ 23
0178    28      PCH(2-I/2)=J
0179            IF(J.NE.7)GØ TØ 26
0180            DØ 29 J2=1,MAXACD
```

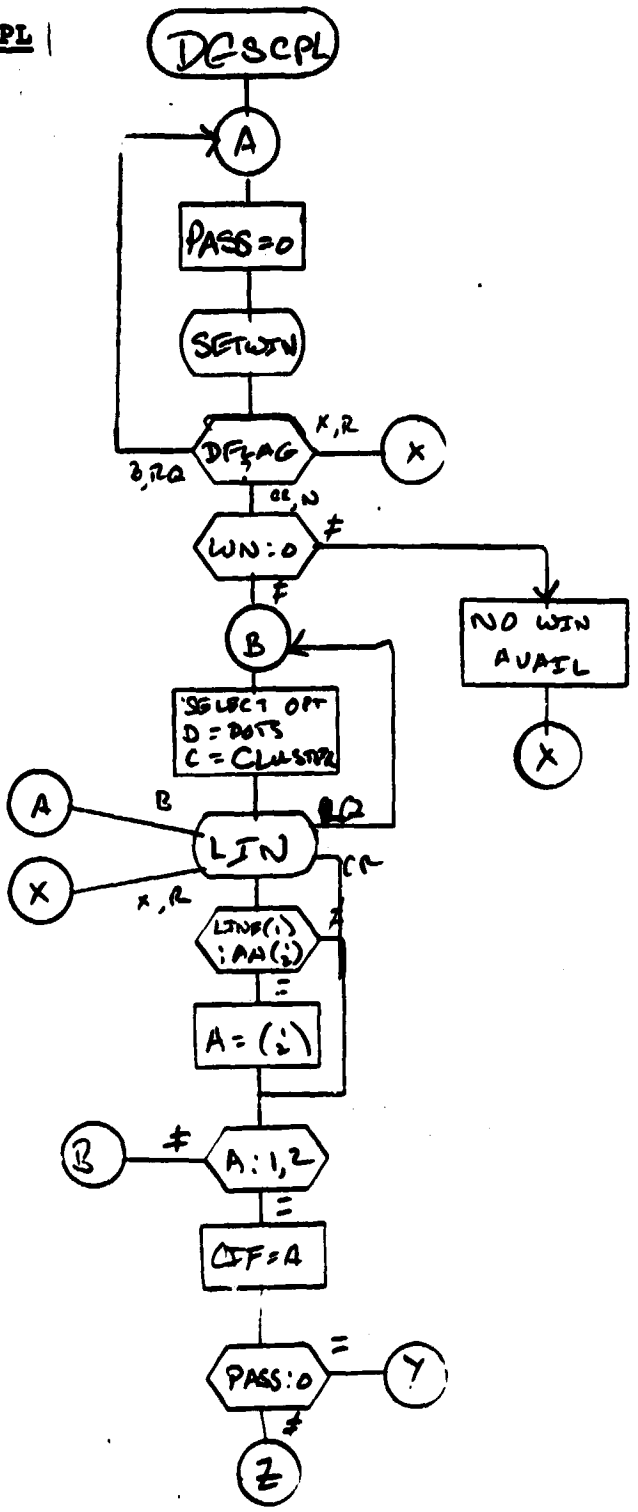
```
0181            IF(ADATES(1,J2),NE,DATE(1))GØ TØ 29
0182            IF(ADATES(2,J2),EQ,DATE(2))GØ TØ 30
0183    29      CØNTINUE
0184              IF(TI,EQ,I0)WRITE(I0,1012) '
0185    1012    FØRMAT(' NØ SØIL GREEN NUMBER FØR THIS ACQUISITIØN.  CHANNEL N '
               1/' THEREFØRE SAME AS CHANNEL G.')
0186            GNU = 0
0187            GØ TØ 26
0188    30      GNU = SØILGR(J2)
0189    26      CØNTINUE
0190              HC = LINE(1)
0191            VC = LINE(3)
0192            CALL VALCK
0193            IF(VAL,EQ,0) GØ TØ 23
0194    31      IF(TI,EQ,I0)WRITE(I0,1013)SCA
0195    1013    FØRMAT('$ENTER SCALING: (F)IXED, (G)LØBAL, ØR (A)NALYST INPUT ',
                1 '  ',A1,' >')
0196            CALL LIN(I0,LINE,I',DFLAG)
0197            GØ TØ (100,23,100,31,33,32)DFLAG
0198    33        LINE(1)=SCA
0199    32      SCA = LINE(1)
0200              GØ TØ (34,35,36)LINE(1)-(LINE(1)/4)*4
0201            GØ TØ 31
0202    35      VMIN = FUL(1,PCH(1))
0203            VMAX = FUL(2,PCH(1))
0204            HMIN = FUL(1,PCH(2))
0205            HMAX = FUL(2,PCH(2))
0206            GØ TØ 37
0207    36      VMIN = GMIN
0208            VMAX = GMAX
0209            HMIN = GMIN
0210            HMAX = GMAX
0211            GØ TØ 37
0212    34      IF(TI,EQ,I0)WRITE(I0,1014)VMIN,VMAX
0213    1014    FØRMAT('$ENTER MINIMUM AND MAXIMUM FØR VERTICAL AXIS,  ',I3,' ,',I
               13,'  >')
0214            CALL LIN(I0,LINE,TI,DFLAG)
0215            GØ TØ (100,31,100,34,38,39)DFLAG
0216    39      I = 0
0217            CALL INTFF(I,LINE,74,VMIN)
0218            CALL INTFF(I,LINE,74,VMAX)
0219    38      IF(VMIN,GE,VMAX)GØ TØ 341
0220            IF(VMIN,GE,-5,AND,VMAX,LE,128) GØ TØ 399
0221    341       WRITE(I0,1018)
0222            TI = I0
0223            GØ TØ 34
0224    399       IF(TI,EQ,I0)WRITE(I0,1015)HMIN,HMAX
0225    1015    FØRMAT('$ENTER MINIMUM AND MAXIMUM FØR HØRIZØNTAL AXIS,  ',I3,' ,
               1',I3,' >')
0226            CALL LIN(I0,LINE,TI,DFLAG)
0227            GØ TØ (100,34,100,399,41,40)DFLAG
0228    40      I = 0
0229            CALL INTFF(I,LINE,74,HMIN)
0230            CALL INTFF(I,LINE,74,HMAX)
0231    41      IF(HMIN,GT,HMAX) GØ TØ 398
0232            IF(HMIN,GE,-K,AND,HMAX,LE,128) GØ TØ 37
```

```
0233     398      WRITE(I0,1018)
0234              TI = I0
0235              GO TO 399
0236     37       XYI(1)=YI-2
0237              XYI(2)=XI+2
0238              XYA(1)=YI+2
0239              XYA(2)=XA-1
0240              0XY(1)=VMIN
0241              0XY(2)=HMIN
0242              MIN(1)=YA+1
0243              MIN(2)=XI+1
0244              SXY(1)=FLOAT((YA-YI+3))/(VMAX-VMIN)
0245              SXY(2)=FLOAT((XA-XI+3))/(HMAX-HMIN)
0246              WRITE(I0,1016)VMIN,VMAX,HMIN,HMAX
0247     1016     FORMAT(' SELECTED SCALE:  VMIN = ',I3,' VMAX = ',I3,' HMIN = ',I3,
                1' HMAX = ',I3/)
0248     12       IF(TI.EQ.I2)WRITE(I0,1017)TH
0249     1017     FORMAT(' SELECT OUTPUT THEME NUMBER,  ',I1,' >')
0250              CALL LIN(I0,LINE,TI,DFLAG)
0251              GO TO (100,42,100,12,44,43)DFLAG
0252     42       IF(PASS.EQ.0) GO TO 31
0253              GO TO 10
0254     43       I = 0
0255              CALL INTFF(I,LINE,74,TH)
0256     44       IF(TH.GT.0.AND.TH.LT.9) GO TO 45
0257              WRITE(I0,1018)
0258     1018     FORMAT(' RANGE CHECK'//)
0259              TI = I2
0260              GO TO 12
0261     1234     CONTINUE
0262              WRITE(I0,1235)
0263     1235     FORMAT(' NO DOTS SELECTED,  TRY AGAIN...'//)
0264     45       IF(CIF.EQ.1)CALL DATIN(I0,II)
0265              IF(CIF.EQ.2)CALL CLUSEL(I0,II,LIST,ND)
0266                IF(ND.EQ.0.AND.CIF.EQ.2)GO TO 1234
0267                IF(NUMDOT.EQ.0.AND.CIF.EQ.1) GO TO 1234
0268              GO TO (451,100,12)II
0269     451      IF(CIF.NE.1)GO TO 50
0270              ND = NUMDOT
0271              DO 49 I = 1,ND
0272     49       LIST(I)=DOTARY(I)
0273                GO TO 46
0274     50       CONTINUE
0275              OPEN(UNIT=1,NAME='[300,1]CLUSTATS.TMP',ACCESS='DIRECT'
                1 ,TYPE = 'OLD',RECORD SIZE=36,MAXREC=MAXSUB,READONLY)
0276              RECL = 144
0277                WRITE(LP,2001)
0278              GO TO 47
0279     46       OPEN(UNIT=1,NAME='[300,1]DOTS.TMP',ACCESS='DIRECT',
                1 TYPE = 'OLD',RECORDSIZE=13,MAXREC=NDOTS,READONLY)
0280              RECL = 52
0281                WRITE(LP,2000)
0282     47       CONTINUE
         D        WRITE(LP,5433)XYI,XYA,0XY,SXY,XI,XA,YI,YA,VMIN,VMAX,HMIN,HMAX
         D5433    FORMAT(' ',8(F10.3,X)/' ',8(I5,X))
0283              I = 0
```

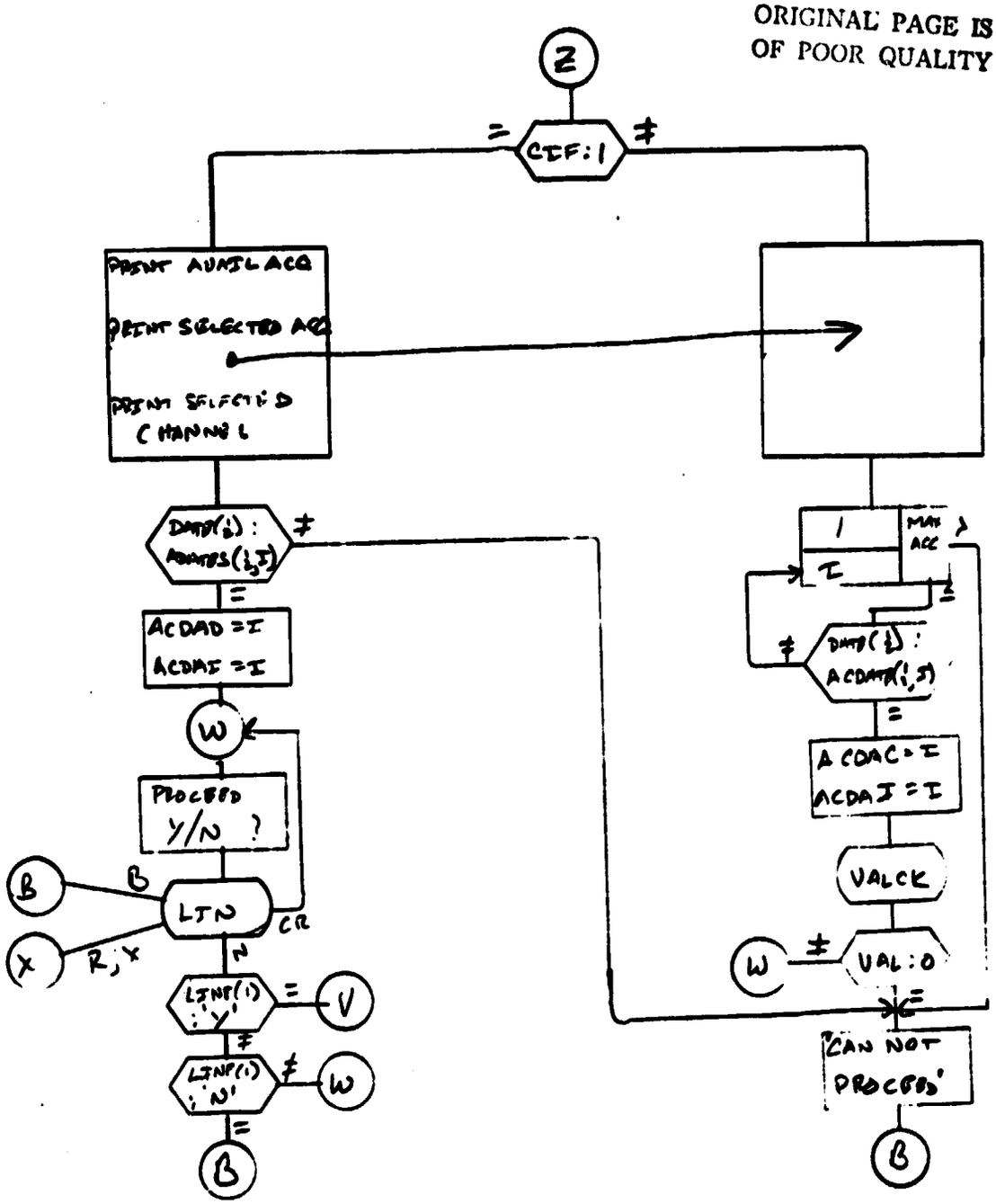
```
0284            DØ 48 I6 = 1,ND
0285          J = LIST(I6)
0286            IF(J.GT.NØSUB.AND.CIF.EQ.2)GØ TØ 48
0287            I = I + 1
0288          PTR(I)=5
      D         WRITE(LP,5431)I,J,RECL,PTR(I),LIST(I),ACDAJ,ACDAI
      D5431     FØRMAT((' ',16(03,2X)))
0289          READ(1'J)(INRAY(K),K=1,RECL)
      D         WRITE(LP,5431)(INRAY(K),K=1,RECL)
0290          DB(1,I)=J
0291          IF(CIF.NE.1) GØ TØ 51
0292      200 DØ 52 J = 1,MAXCHN
0293          DB (J+1,I)=0
0294          IF(III(J).EQ.0) GØ TØ 52
0295          K = III(J)
0296          A = FFUNC(DØTS(K,ACDAJ))
0297          DB(J+1,I)=(FLØAT(G(J))/100.)*(A+FLØAT(B(J))/100.)
0298      52  CØNTINUE
0299          DØ 53 J = 1,2
0300          K = PCH(J)
0301            GØ TØ (54,56)(((K-1)/2)*2)/3
0302          A = FFUNC(DØTS(K,ACDAI))
0303          GØ TØ 55
0304      54    A = DØTSW(K-2,ACWAI)
0305          GØ TØ 55
0306      56  A = DØTSW(3,ACDAI)-GNU
0307      55    IF(J.EQ.1)F=A
0308            DBW(6-J,I)=(A-ØXY(J))*SXV(J)+XYI(J)
0309            IF(DBW(6-J,I).LT.MIN(J))DBW(6-J,I)=MIN(J)
0310          IF(DBW(6-J,I).GT.XYA(J))DBW(6-J,I)=XYA(J)
0311      53  CØNTINUE
0312            IF = F
0313            IA =A
0314            WRITE(LP,2002)FFUNC(DB(1,I)),IA,IF
0315          GØ TØ 48
0316      51  DØ 61 J=1,2
0317          K = PCH(J)
0318          GØ TØ (57,58,59)(((K-1)/2)*2)/3+1
0319      57  F = CLUS(K,ACDAI)
0320          GØ TØ 60
0321      58  F = CLUSW(12+K,ACDAI)
0322          GØ TØ 60
0323      59  F=CLUSW(17,ACDAI)-GNU
0324      60    IF(J.EQ.1)A=F
0325            DBW(6-J,I)=(F-ØXY(J))*SXV(J)+XYI(J)
0326            IF(DBW(6-J,I).LT.MIN(J))DBW(6-J,I)=MIN(J)
0327          IF(DBW(6-J,I).GT.XYA(J))DBW(6-J,I)=XYA(J)
0328      61    CØNTINUE
0329            WRITE(LP,2003)FFUNC(DB(1,I)),F,A
0330      48  CØNTINUE
0331          CLØSE(UNIT=1)
0332            ND = I + 1
0333            DBW(4,ND)=XI
0334            DBW(5,ND)=YI-1
0335            PTR(ND)=5
      D         WRITE(LP,5431)ACDAJ,ACDAI
```

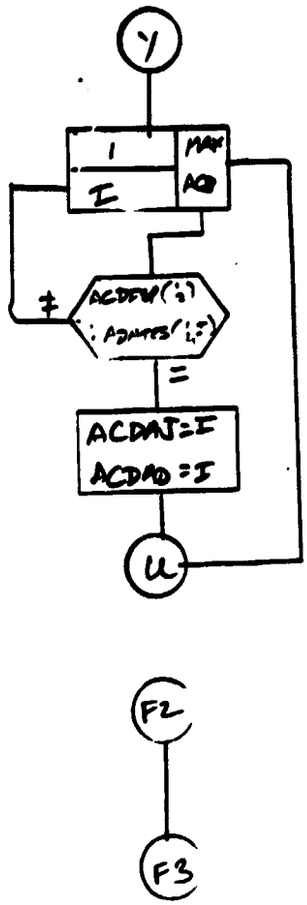
```
FØRTRAN IV-PLUS V02-04          14104116    31-AUG-77          PAGE 8
DØSCPL.FTN      /TR:BLØCKS/WR
        D         WRITE(LP,5432)((D2(I,J),I=1,5),(DBW(I,J),I=4,5),PTR(J),J=1,ND)
0336              CALL SHELL(ND,5,DBW,PTR)
        D         WRITE(LP,5431)ACDAJ,ACDAI
        D         WRITE(LP,5432)((D2(I,J),I=1,5),(DBW(I,J),I=4,5),PTR(J),J=1,ND)
        D5432     FØRMAT(' ',5(Ø3,2X),I5,2X,I5,2X,I5)
0337              CLØSE(UNIT=LP)
0338              J = 1
0339              K = YA+1
0340              CALL SETVID(YA,255)
        D         WRITE(LP,5431)ACDAJ,ACDAI
0341              DØ 62 I = 1,ND
0342              IF(DBW(5,I).LE.DBW(5,J)) GØ TØ 62
0343              CALL PLØT(K,I-1)
        D         WRITE(LP,5431)ACDAJ,ACDAI
0344              J = I
0345    62          CØNTINUE
0346    621         CALL PLØT(K,I-1)
        D         WRITE(LP,5431)ACDAJ,ACDAI
0347              CALL SETVID(YI,255)
        D         WRITE(LP,5431)ACDAJ,ACDAI
0348              CALL SETVID(YI-1,255)
        D         WRITE(LP,5431)ACDAJ,ACDAI
0349              ND = ND - 1
0350              CALL LRJUNK
        D         WRITE(LP,5431)ACDAJ,ACDAI
0351              IF(TI.NE.IØ) GØ TØ 622
0352    2000    FØRMAT('1SCATTER PLØT REPØRT!'/' DØT GRID NUMBER    X    Y')
0353    2001    FØRMAT('1SCATTER PLØT REPØRT!'
               1/' CLUSTER NUMBER     X    Y')
0354    2002      FØRMAT(' ',8X,I3,6X,I3,I4)
0355    2003      FØRMAT(' ',6X,I3,4X,2F5.1)
0356    622       CØNTINUE
0357              PASS = 1
0358    63        IF(TI.EQ.IØ)WRITE(TØ,1089)
        D         WRITE(LP,5431)ACDAJ,ACDAI
0359    1088    FØRMAT('$FØR NEXT SCATTER PLØT, (S)AME, ØR (N)EW WINDØW?  >')
0360              CALL LINEIØ,LINE,TI,DFLAG)
        D         WRITE(LP,5431)ACDAJ,ACDAI
0361              GØ TØ (100,49,100,43,100,64)DFLAG
0362    64        GØ TØ (1,3)(LINE(1)-76)/5+1
0363    100       WRITE(TØ,1019)
0364    1019    FØRMAT('$(R)ESTART ØR E(X)IT?  >')
0365              CALL LINEIØ,LINE(1),TI,DFLAG)
0366              GØ TØ (101,63,1,100,100,100
               1 )DFLAG
0367    101       CØNTINUE
0368                CALL ELAPSE(2)
0369                IF(RECL.EQ.0)WIN(1,WN)=0
0370              RETURN
0371              END
```

12.1 SUBROUTINE DGSCPL



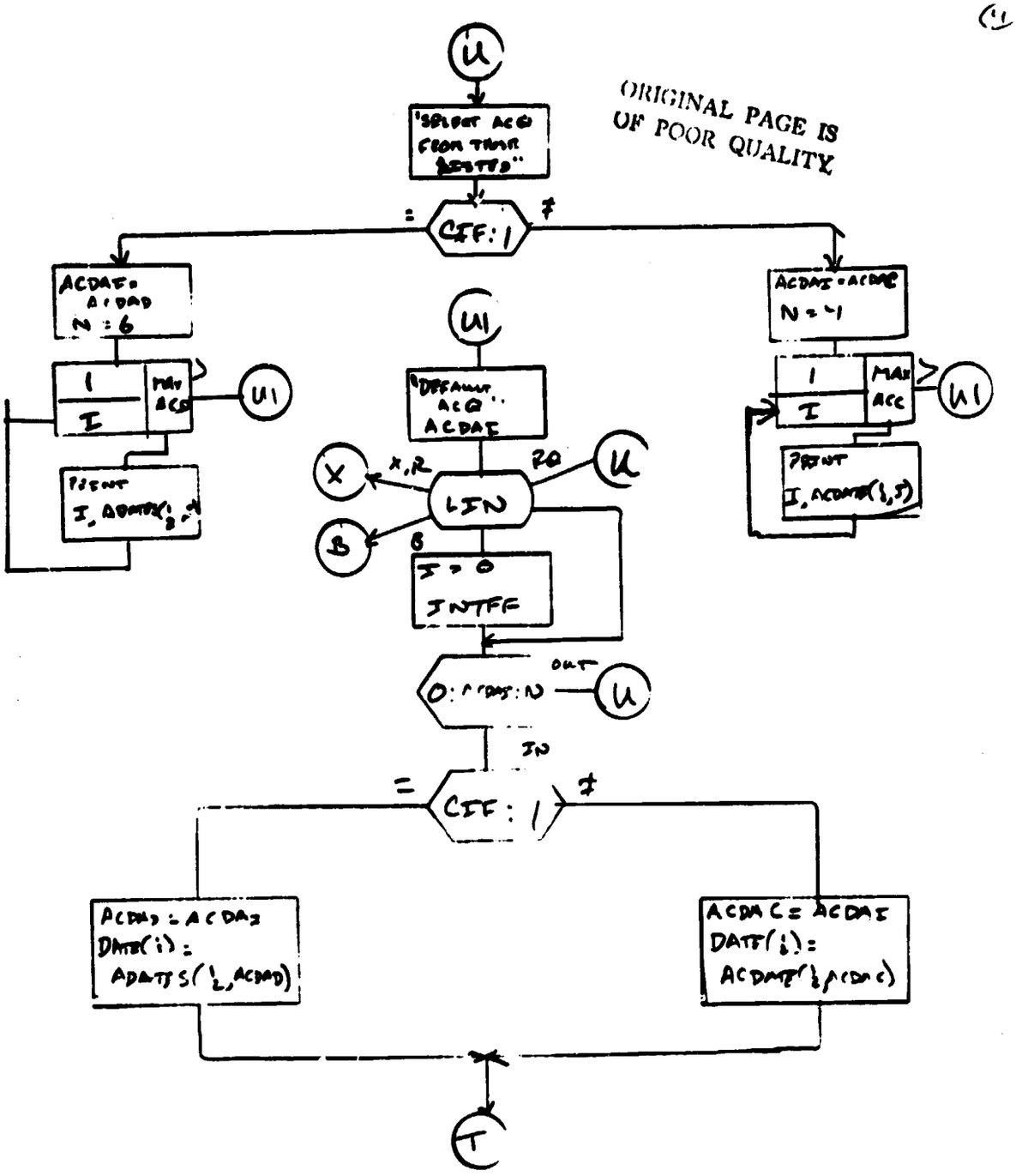
ORIGINAL PAGE IS
OF POOR QUALITY

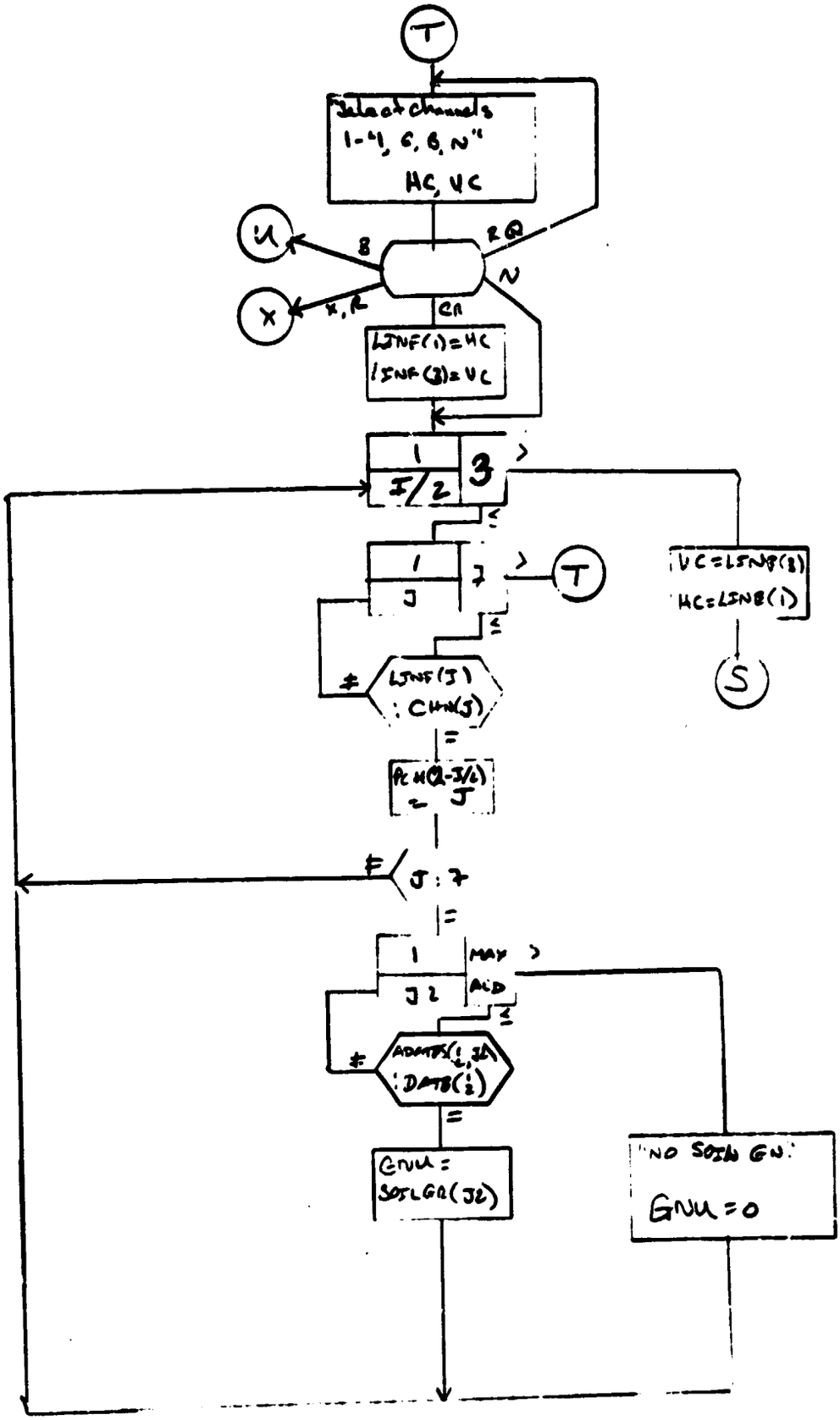


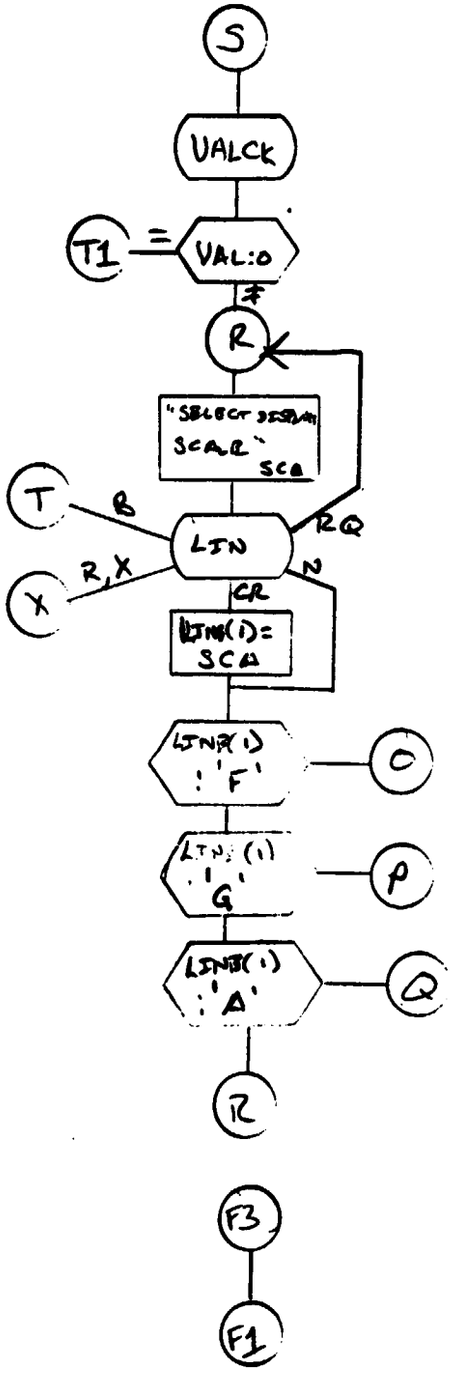


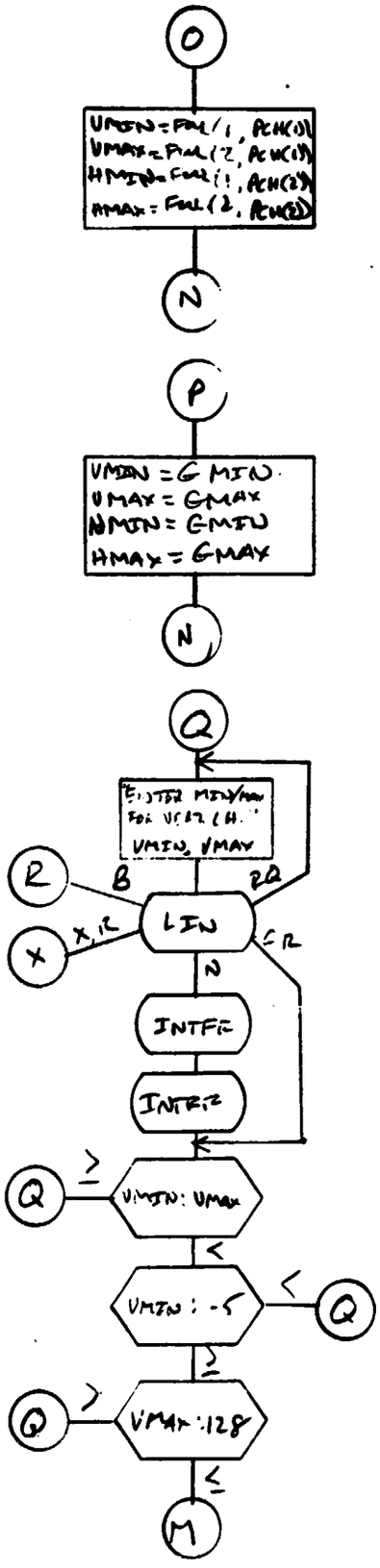
ORIGINAL PAGE IS
OF POOR QUALITY

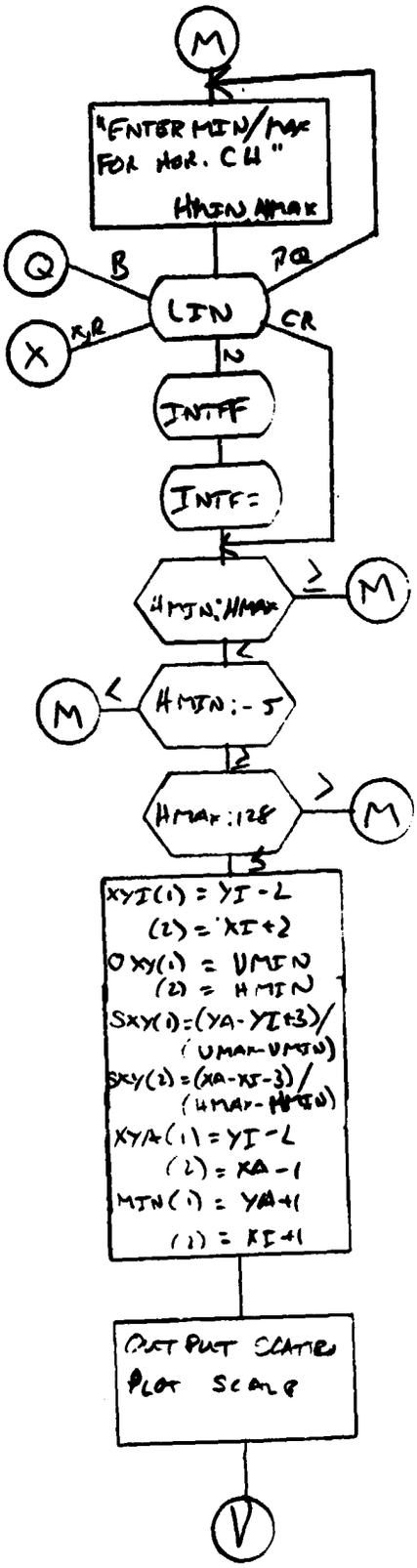
ORIGINAL PAGE IS
OF POOR QUALITY

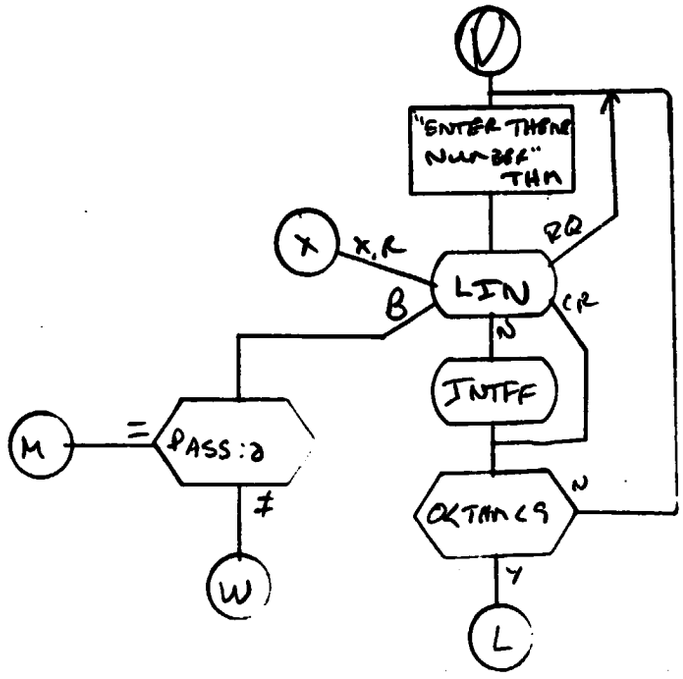


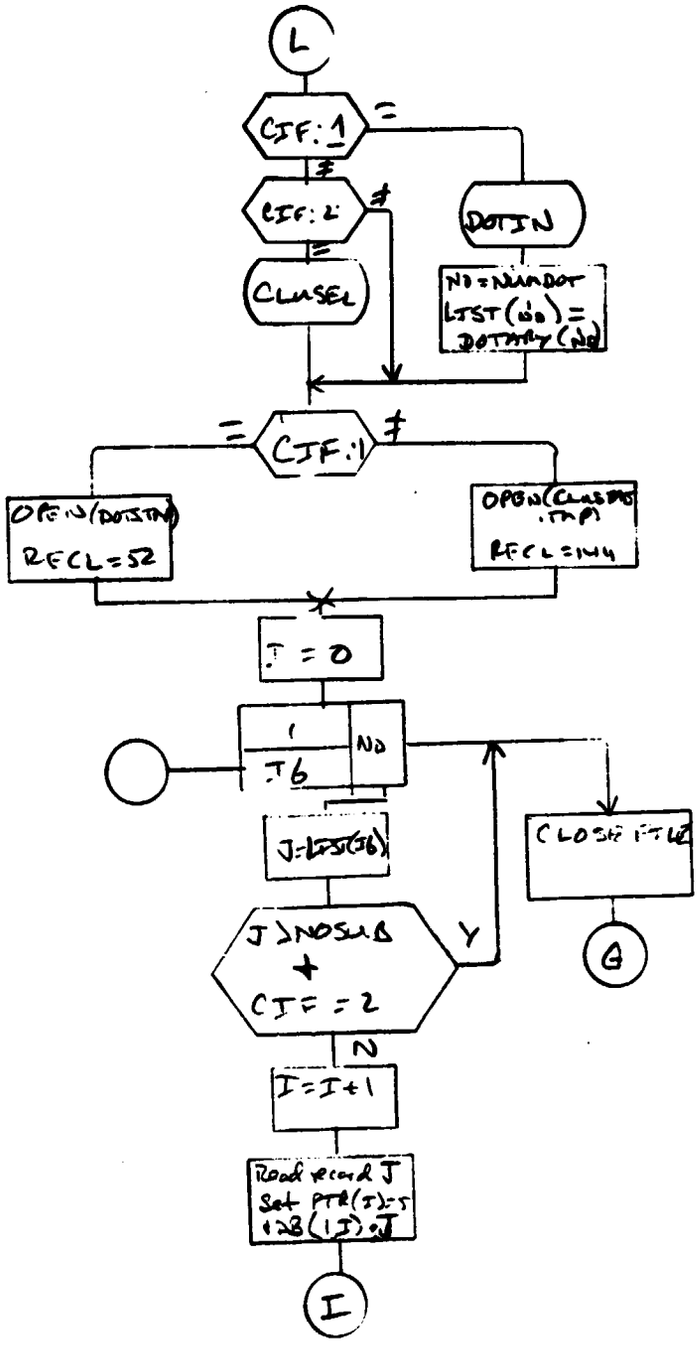

T
Select channels
1-4, 6, 8, N"
HC, VC
U
X
B
RQ
N
R, R
CA
LINF(1)=HC
LINF(2)=VC
1
I/2
3
1
J
2
T
J
LINF(J)
: CHN(J)
VC=LINB(2)
HC=LINB(1)
S
PCHN(2-I/2)
= J
J : 2
1
J 2
MAX
ADD
ADDRBS(1, J2)
: DATB(1, 2)
GNUM =
SOILGR(J2)
"NO SOIL GN."
GNUM = 0

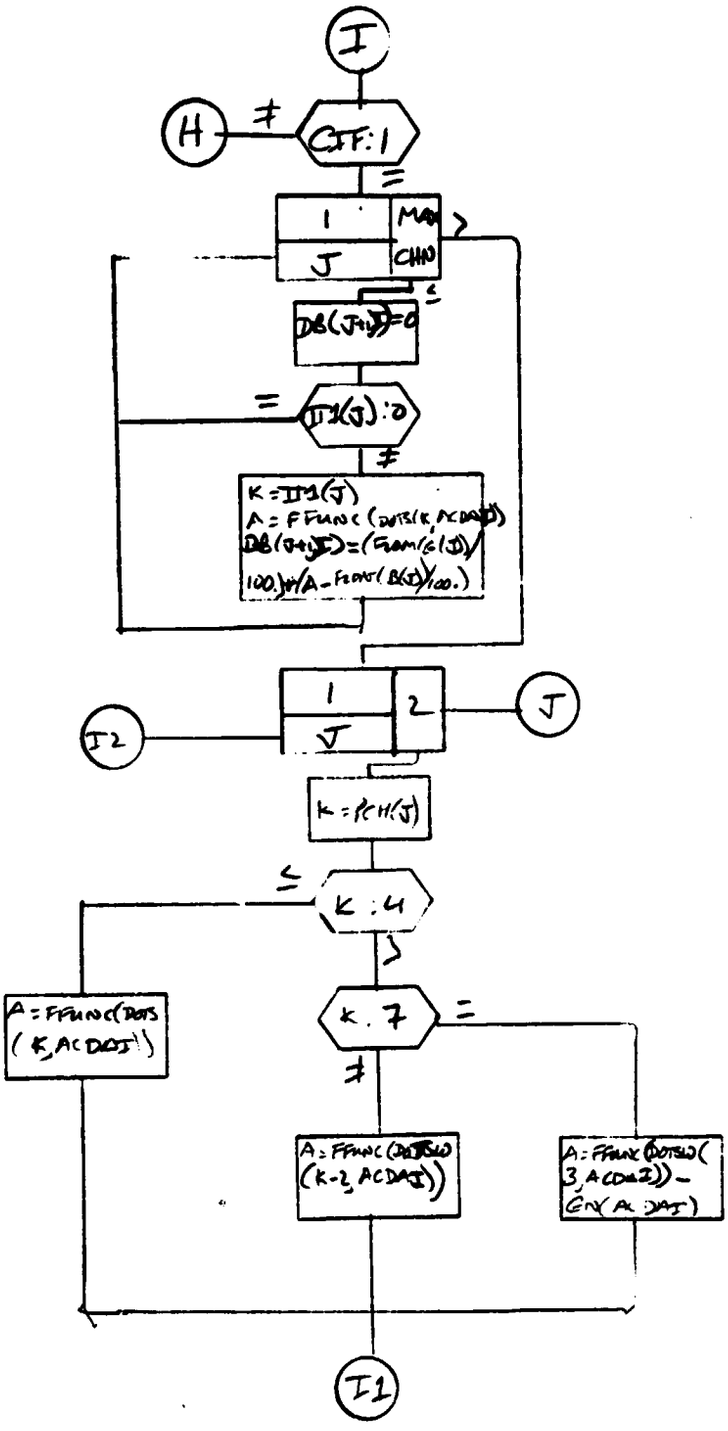

S
VALCK
T1
=
VAL:0
≠
R
"SELECT SCALAR" SCA
T
B
LIN
RQ
X
R,X
N
CR
LIN(I)= SCA
LIN(I) ! 'F'
C
LIN(I) 'G'
P
LIN(I) 'A'
Q
R
F3
F1

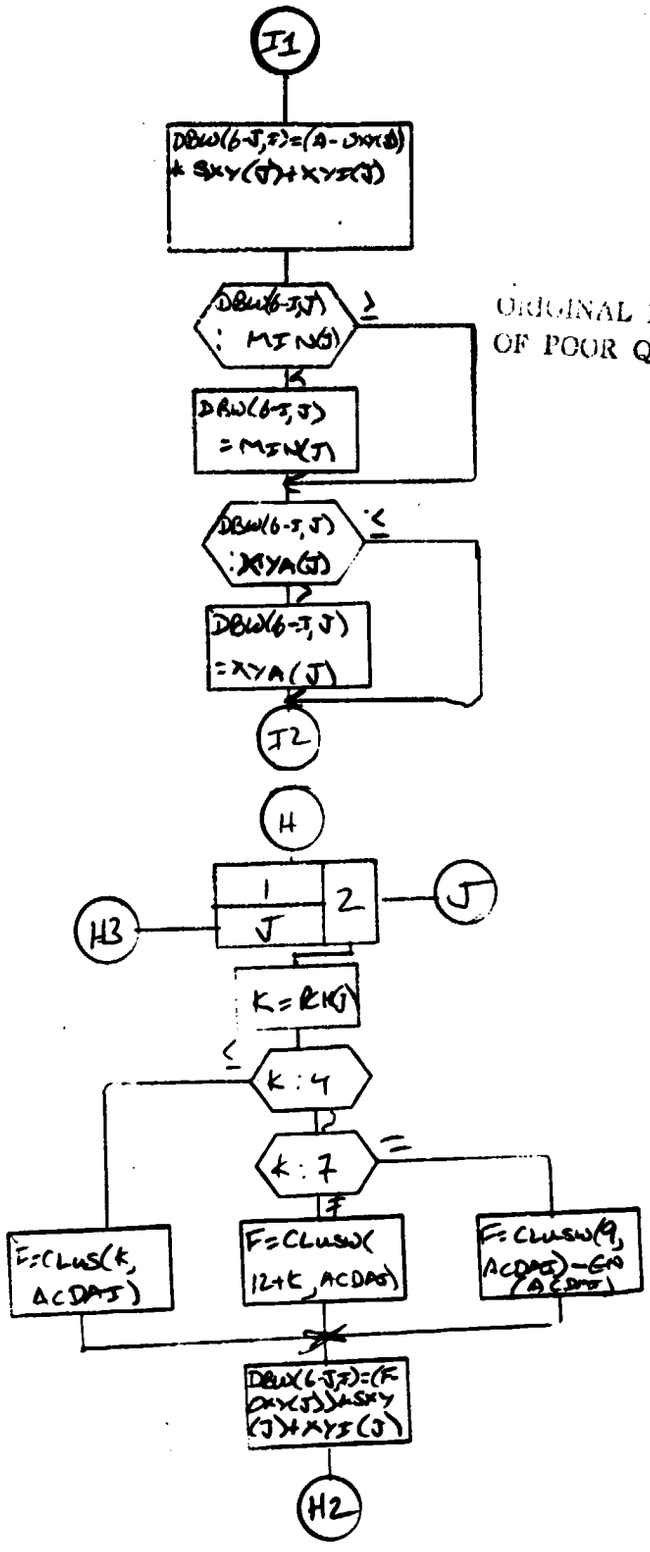

O
VMIN = FUNC(1, ACHC(0))
VMAX = FUNC(2, ACHC(1))
HMIN = FUNC(1, ACHC(2))
HMAX = FUNC(2, ACHC(3))
N
P
VMIN = GMIN
VMAX = GMAX
HMIN = GMIN
HMAX = GMAX
N
Q
ENTER MIN/MAX FOR VFB2 (H.
VMIN, VMAX
R
B
2Q
LIN
X
X, R
≤ R
N
INTFC
INTFC
Q
≥
VMIN : VMAX
<
VMIN : -5
<
Q
≥
Q
>
VMAX : 128
≤
M


M
"ENTER MIN/MAX FOR HOR. CH"
HMIN,HMAX
Q
B
?Q
LIN
X
?X
CR
N
INTFF
INTF=
HMIN:HMAX
≥
M
<
M
HMIN:-5
HMAX:128
>
M
XYI(1) = YI-2
(2) = XI+2
OXY(1) = VMIN
(2) = HMIN
SXY(1)=(YA-YI+3)/(VMAX-VMIN)
SXY(2)=(XA-XI-3)/(HMAX-HMIN)
XYA(1) = YI-2
(2) = XA-1
MIN(1) = YA+1
(2) = XI+1
OUTPUT SCATTER PLOT SCALE
V

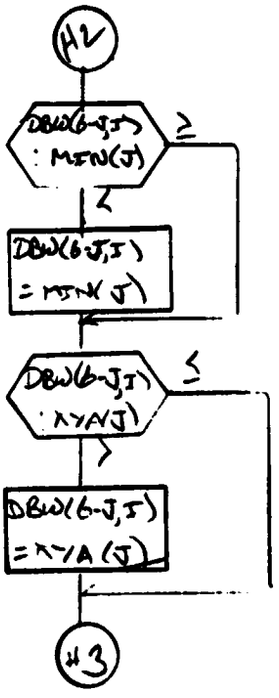

"ENTER THEIR NUMBER" THEN
X
X,R
RQ
LIN
B
CR
N
INTFF
M
=
PASS:0
≠
OKTHEN?
N
Y
W
L


L
CIF:1
=
≠
CIF:2
≠
=
DOTIN
CLUSEL
NO=NUMDOT
LIST(NO)=
DOTARY(NO)
=
CIF:1
≠
OPEN(DOTIN)
RECL=52
OPEN(CLUSFIL
.TMP)
RECL=1144
I=0
I
I6
NO
J=LIST(I6)
CLOSE FILE
J>NOSUB
+
CIF=2
Y
N
G
I=I+1
Read record J
Set PTR(I)=J
+2B(1J)=J
I

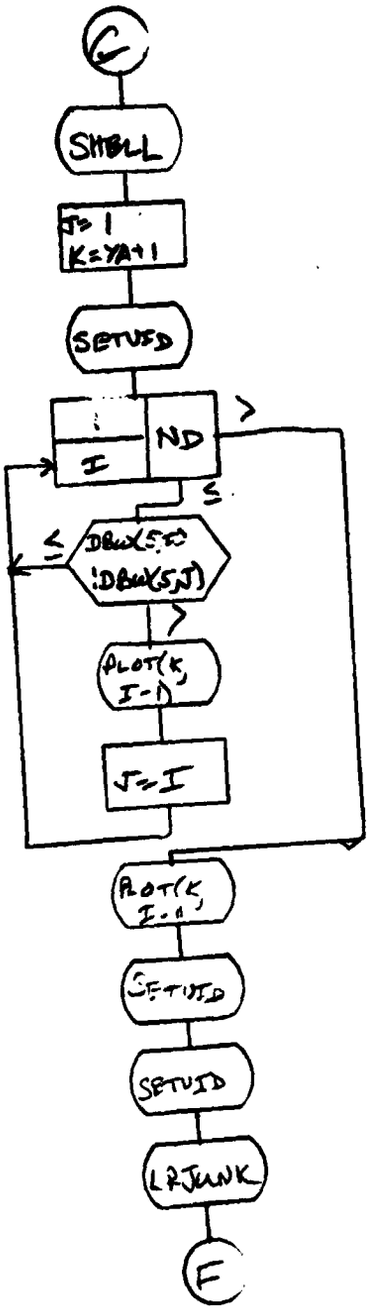

I
H
≠
CTF: 1
=
1
J
MAX
CHN
>
≤
DB(J+1,J)=0
=
II2(J):0
≠
K=II1(J)
A=FFUNC(DOTS(K,ACDAT))
DB(J+1,J)=(FLOAT(G(J))
100.)*A=FLOAT(B(J))/100.)
1
J
2
J
I2
K=PCH(J)
≤
K:4
>
A=FFUNC(DOTS
(K,ACDAT))
K.7
=
≠
A=FFUNC(DOTS(
(K-2,ACDAT))
A=FFUNC(DOTS(
3,ACDAT))-
GRV(ACDAT)
I1



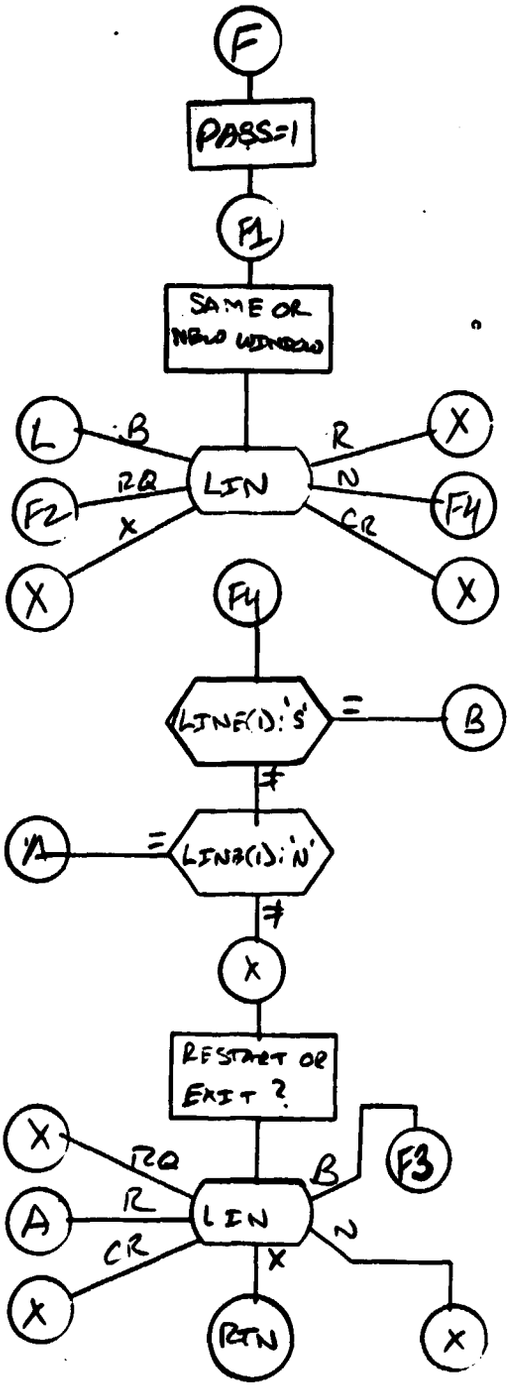
ORIGINAL PAGE IS
OF POOR QUALITY



ORIGINAL PAGE IS
OF POOR QUALITY


C
SYBLL
J=1
K=YA91
SETUSD
I
ND
>
≤
≤
DBW(5,I)
!DBW(5,J)
>
PLOT(K,
I-1)
J=I
PLOT(K,
I-1)
SETUSD
SETUSD
LRJUNK
F

(15)


F
PASS=1
F1
SAME OR NEW WINDOW
L
B
R
X
F2
RQ
LIN
N
F4
X
CR
X
F4
LINE(1):'S'
=
B
≠
'L'
=
LINE(1):'N'
≠
X
RESTART OR EXIT ?
X
RQ
B
F3
A
R
LIN
N
CR
X
X
RTN
X

12.2 SUBROUTINE LRJUNK

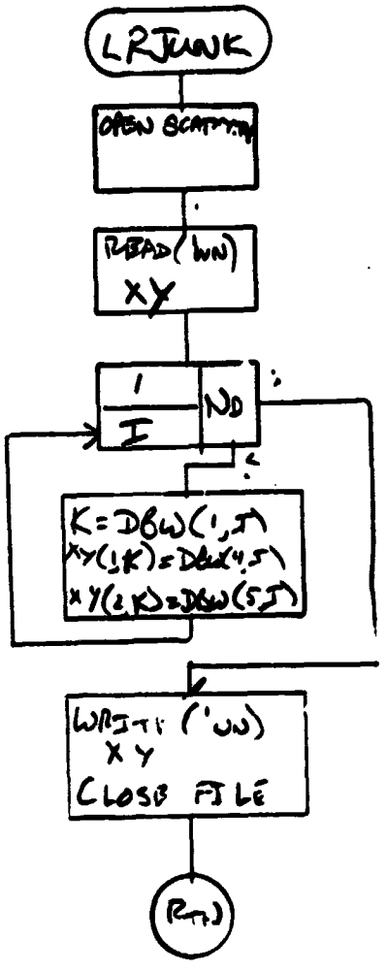
```
MFORTRAN IV-PLUS V02-04          14:09:34   31-AUG-77          PAGE 1
LRJUNK.FTN      /TRIBLOCKS/WR
0001            SUBROUTINE LRJUNK
0002            INCLUDE 'TOP.INC'
      *  C!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!C
      *  C                                                                      C
      *  C                                                                      C
      *  C                                                                      C
      *  C                                                                      C
         C                WRITE SCATXY FILE                                     C
         C                T. KELL/LEC/4/77                                      C
         C                LRJUNK.FTN                                            C
0003            INCLUDE 'BOT.INC'
      *  C                                                                      C
      *  C                                                                      C
      *  C                                                                      C
      *  C                                                                      C
      *  C!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!C
0004            INCLUDE 'HUGSYSEAR:COM'
0005 *            INCLUDE '[300,3]CAMSCOM20.INC'
0006 *          INCLUDE 'SYI[300,3]CAMSPARAM.INC'
0007 *          PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      *         1,MAXV=11,NDOTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *         2NOSPWD=6,NODTWD=10
0008 *          EQUIVALENCE (C1,ACDATE),(C2,ISEG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0009 *          INTEGER C1(469),C2(256),C3(71),C4(348),C5(629)
      *  C*
0010 *          INTEGER ACDATE,SUBCAT,SUBPOP,CATNAT,CATTH
0011 *          BYTE CHANVEC,NOCHAN,NOSUB,DOTCAT,DOTCLU
0012 *          COMMON/COM1/ACDATE(2,MAXACD),CHANVEC(MAXCHN,MAXACC),NOCHAN,NOSUB,
      *         1SUBCAT(MAXSUB),SUBPOP(MAXSUB),CATNAT(MAXCAT),CATTH(MAXCAT),NODO,
      *         2NODU,NOTH,DOTCAT(NDOTS),DOTCLU(NDOTS)
      *  C*
0013 *          INTEGER ADATES,SUNAZ,ANALST,FLDDAY,DOTDAY,PDATE1,TDATE1
0014 *          INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RANDOM,GRID
0015 *          BYTE DELFLG,NOACQ,SOILGR,SUNEL,ASTART,NTYPE1,ALP,ALPO
0016 *          BYTE POTCT,POTCTO,VAR,VARO,DLABEL,TYPE
0017 *          COMMON/COM2/ISEG,DELFLG,NOACQ,ADATES(2,MAXACD),SOILGR(MAXACD),
      *         1SUNEL(MAXACD),SUNAZ(MAXACD),IMDATE(2),ANALST(5),FLDDAY(2),
      *         2DOTDAY(2),ASTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      *         3PDATE3(2),TDATE3(2),NOCAT,CATNAM(MAXCAT),ALP(MAXCAT),ALPO,
      *         4       POTCT(MAXCAT),POTCTO,VAR(MAXCAT),VARO
      *  C*
0018 *          INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      *         1UFLAG4
0019 *          INTEGER PFLAG,DSKMNT
0020 *          COMMON/COM3/PFLAG,DSKMNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
      *         1,UFLAG2,UFLAG3,UFLAG4,NEWLAB(MAXSUB)
      *  C*
0021 *          INTEGER TX1,TY1,TX2,TY2,ACDISP,G,B,DTWIND,DOTARY,GMIN,GMAX,FUL
0022 *          INTEGER SPWIND,CLAWND,CLUWND
0023 *          COMMON/COM4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACDISP(2),I1(4),G(4),
      *         1B(4),DTWIND(5,NODTWD),SPWIND(5,NOSPWD),IMWIND(4),NUMDOT,
      *         2DOTARY(NDOTS),GMIN,GMAX,FUL(2,7),CLAWND(8),CLUWND(8)
0024 *          COMMON/COM5/DISKID,RANDOM(NDOTS),GRID(NDOTS),DLABEL(NDOTS),
      *         1TYPE(NDOTS),RECLOC
0025 *            BYTE ACDAC,ACDAI,ACDAJ,ACDAD,BUF,DB,LINE,VAL
0026 *            BYTE PASS,PCH,THS
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
FORTRAN IV-PLUS V02-04          14:05:34    31-AUG-77          PAGE 2
LRJUNK.FTN      /TRIBLOCKS/WR
0027 *          INTEGER CIF,DFLAG,DATE,HMIN,HMAX,I,J,K,LIST,ND,TH,VMIN
0028 *          INTEGER VMAX,XA,XI,YA,YI,TI
0029 *          INTEGER PTR,RECL,WN
0030 *          REAL DXY,SXY,XYI,XYA
0031 *          COMMON /HUGGY/ ACDAC,ACDAI,ACDAJ,ACDAD,BUF(512,4),CIF,
     *         1 DFLAG,DATE(2),DB(10,ND0TS+1),HMIN,HMAX,I,J,K,LINE(74),
     *         2 LIST(ND0TS),ND,VAL,PASS,DXY(2),PCH(2),PTR(ND0TS+1),RECL,
     *         3 WN,SXY(2),TH,THB(64),VMIN,VMAX,XA,XI,XYA(2),XYI(2),YA,YI,TI
0032 *          INTEGER DBW(5,ND0TS+1)
0033 *          EQUIVALENCE(DB(1,1),DBW(1,1))
0034 *          BYTE INRAY(360*MAXACC),D0TS(8,MAXACD),D0TP(6)
0035 *          INTEGER CLUSW(18,MAXACC),D0TSW(14,MAXACD)
0036 *          REAL CLUS(9,MAXACC)
0037 *          EQUIVALENCE (BUF(1,1),INRAY(1))
0038 *          EQUIVALENCE(INRAY(1),D0TP(1))
0039 *          EQUIVALENCE (D0TP(5),D0TS(1,1))
0040 *          EQUIVALENCE (D0TP(5),D0TSW(1,1))
0041 *          EQUIVALENCE (INRAY(1),CLUS(1,1))
0042 *          EQUIVALENCE (CLUS(1,1),CLUSW(1,1))
0043            INTEGER FFUNC
0044            INTEGER XY(2,ND0TS)
0045            EQUIVALENCE (BUF(1,1),XY(1,1))
0046            OPEN (UNIT=1,NAME='[300,1]SCATXY.TMP',TYPE='OLD'
                1 ,ACCESS = 'DIRECT')
0047            READ(1'WN)XY
0048             DO 1 I = 1,ND
0049            K = FFUNC(DB(1,I))
0050            XY(1,K)=DBW(4,I)
0051            XY(2,K)=DBW(5,I)
0052      1     CONTINUE
0053            WRITE(1'WN)XY
0054            CLOSE(UNIT=1)
0055            RETURN
0056            END
```

ORIGINAL PAGE IS
OF POOR QUALITY

12.2 SUBROUTINE LRJUNK



12.3 SUBROUTINE PLOT

```
MEGATRAN IV-PLUS V02-04          14:06:23  31-AUG-77          PAGE 1
PLOT.FTN        /TRIALOCKS/NR
0001            SUBROUTINE PLOT(L,KM)
0002              INCLUDE 'TOP.INC'
      *  C!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!C
      *  C                                                                      C
      *  C                                                                      C
      *  C                                                                      C
      *  C                                                                      C
      *  C                                                                      C
                C                  PLOT DOTS IN SCATTER PLOT                   C
                C                  T. KELL/LEC/4277                             C
                C                  PLOT.FTN                                    C
0003            INCLUDE 'BOI.INC'
      *  C                                                                      C
      *  C                                                                      C
      *  C                                                                      C
      *  C                                                                      C
      *  C!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!C
0004            INCLUDE 'HUGGYBEAR.CR4'
0005  *           INCLUDE '[300,3]CAMSCR4.INC'
0006  *         INCLUDE 'SYS[300,3]CAMSPARAM.INC'
0007  *         PARAMETER MAXCAT=8,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      *        1,MAXV=11,NDOTS=200,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *        2NOSP=0=6,NODTWD=10
0008  *         EQUIVALENCE (C1,ACDATE),(C2,ISEG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0009  *         INTEGER C1(469),C2(256),C3(71),C4(348),C5(620)
      *  C*
0010  *         INTEGER ACDATE,SUBCAT,SUBPOP,CATKNT,CATTH
0011  *         BYTE CHNVEC,NOCHAN,NOSUB,DOTCAT,DOTCLU
0012  *         COMMON/COM1/ACDATE(2,MAXACC),CHNVEC(MAXCHN,MAXACC),NOCHAN,NOSUB,
      *        1SUBCAT(MAXSUB),SUBPOP(MAXSUB),CATKNT(MAXCAT),CATTH(MAXCAT),NODO,
      *        2NOOU,NOTH,DOTCAT(NDOTS),DOTCLU(NDOTS)
      *  C*
0013  *         INTEGER ADATES,SUNAZ,ANALST,FLDDAY,DOTDAY,PDATE1,TDATE1
0014  *         INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RANDOM,GRID
0015  *         BYTE DELFLG,NACO,SAILGR,SUNEL,NSTART,NTYPE1,ALP,ALPO
0016  *         BYTE POTCT,POTCTO,VAR,VARO,DLABEL,TYPE
0017  *         COMMON/COM2/ISEG,DELFLG,NOACC,ADATES(2,MAXACD),SAILGR(MAXACD),
      *        1SUNEL(MAXACD),SUNAZ(MAXACD),IMDATE(2),ANALST(5),FLDDAY(2),
      *        2DOTDAY(2),NSTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      *        3PDATE3(2),TDATE3(2),NOCAT,CATNAM(MAXCAT),ALP(MAXCAT),ALPO,
      *        4          POTCT(MAXCAT),POTCTO,VAR(MAXCAT),VARO
      *  C*
0018  *         INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      *        1UFLAG4
0019  *         INTEGER PFLAG,DSKMNT
0020  *         COMMON/COM3/PFLAG,DSKMNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
      *        1,UFLAG2,UFLAG3,UFLAG4,NEWLAB(MAXSUB)
      *  C*
0021  *         INTEGER TX1,TY1,TX2,TY2,ACDISP,G,B,DTWIND,DOTARY,GMIN,GMAX,FUL
0022  *         INTEGER SPWIND,CLAWND,CLUWND
0023  *         COMMON/COM4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACDISP(2),I11(4),G(4),
      *        1R(4),DTWIND(5,NODTWD),SPWIND(5,NOSPWD),IMWIND(4),NUMDOT,
      *        2DOTARY(NDOTS),GMIN,GMAX,FUL(2,7),CLAWND(8),CLUWND(8)
0024  *         COMMON/COM5/DISKID,RANDOM(NDOTS),GRID(NDOTS),DLABEL(NDOTS),
      *        1TYPE(NDOTS),RECLPC
0025  *           BYTE ACDAC,ACDAI,ACDAJ,ACDAD,BUF,DB,LINE,VAL
0026  *           BYTE PASS,PCH,THR
```

PLØT.FIN /TR:PL@CKS/WP

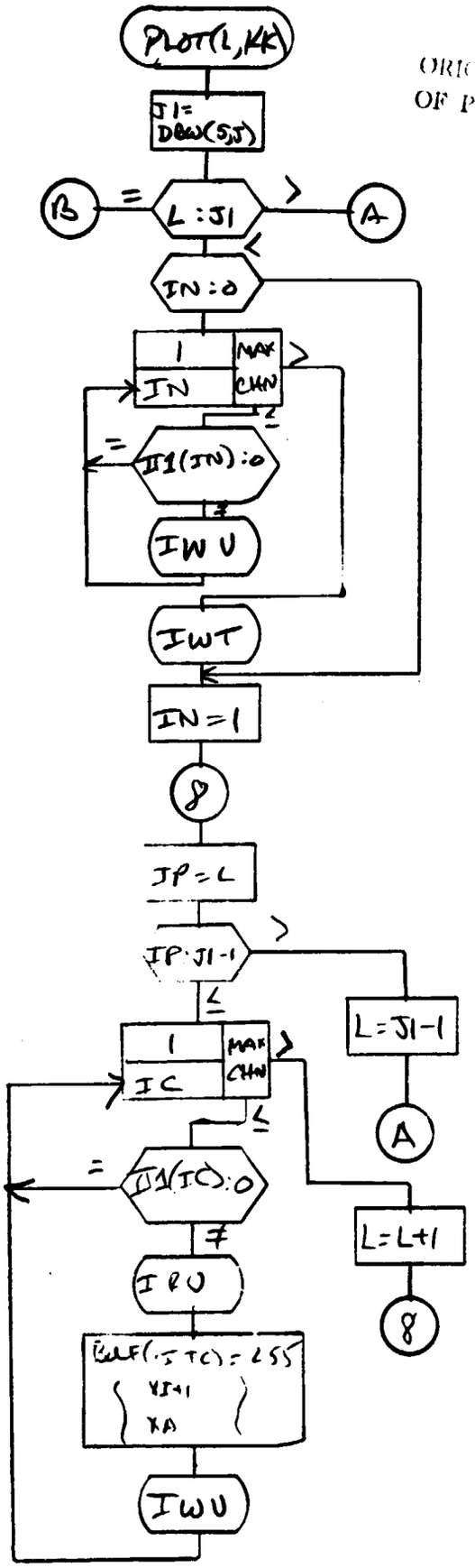
```
0027 *          INTEGER CIF,DFLAG,DATE,HMIN,HMAX,I,J,K,LIST,ND,TH,VMIN
0028 *          INTEGER VMAX,XA,XI,YA,YI,TI
0029 *          INTEGER PTR,RECL,WN
0030 *          REAL ØXY,SYY,XYI,XYA
0031 *          CØMMØN /BUGGY/ ACDAC,ACDAI,ACDAJ,ACDAD,BUF(512,4),CIF,
     *         1 DFLAG,DATE(2),DE(10,ND@TS+1),HMIN,HMAX,I,J,K,LINE(74),
     *         2 LIST(ND@TS),ND,VAL,PASS,ØXY(2),PCH(2),PTR(ND@TS+1),RECL,
     *         3 WN,SXY(2),TH,TØB(64),VMIN,VMAX,XA,XI,XYA(2),XYI(2),YA,YI,TI
0032 *          INTEGER DBW(5,ND@TS+1)
0033 *          EQUIVALENCE(DB(1,1),DBW(1,1))
0034 *          BYTE INRAY(36*MAXACC),D@TS(6,MAXACD),D@TP(6)
0035 *          INTEGER CLUSW(18,MAXACC),D@TSW(4,MAXACD)
0036 *          REAL CLUS(9,MAXACC)
0037 *          EQUIVALENCE (BUF(1,1),INRAY(1))
0038 *          EQUIVALENCE(INRAY(1),D@TP(1))
0039 *          EQUIVALENCE (D@TP(5),D@TS(1,1))
0040 *          EQUIVALENCE (D@TP(5),D@TSW(1,1))
0041 *          EQUIVALENCE (INRAY(1),CLUS(1,1))
0042 *          EQUIVALENCE (CLUS(1,1),CLUSW(1,1))
0043           DATA IN/0/
0044           J1 =DBW(5,J)
0045           IF(L-J1)1,22,3
0046     22      IF(L.EQ.YA+1)G@ T@ 6
0047             G@ T@ 2
0048     1     IF(IN.EQ.0) G@ T@ 4
0049           D@ 5 IN = 1,MAXCHN
0050           IF(II1(IN).EQ.0) G@ T@ 5
0051           CALL INV(IN,L,BUF(1,IN))
0052     5     CØNTINUE
0053             CALL INT (TH,L,1,THB)
0054     4     IN = 1
0055     8     IP = L
0056             IF(IP.GE.YI-1)G@ T@ 19
0057           IF(IP.GT.J1-1) G@ T@ 6
0058           D@ 7 IC = 1,MAXCHN
0059           IF(II1(IC).EQ.0) G@ T@ 7
0060           CALL IRV(IC,IP,BUF(1,IC))
0061           CALL WAIT
0062           BUF(XI+1,IC) = 255
0063           BUF(XI+2,IC)=255
0064           BUF(XA+1,IC)=255
0065           CALL INV(IC,IP,BUF(1,IC))
0066     7     CØNTINUE
0067           L = L + 1
0068           G@ T@ 8
0069     6     L = J1-1
0070     3     D@ 9 IP = 1,MAXCHN
0071           IF(II1(IP).EQ.0) G@ T@ 9
0072           CALL IRV(IP,J1,BUF(1,IP))
0073     9     CØNTINUE
0074           CALL IRT(TH,J1,1,THB)
0075           CALL WAIT
0076           D@ 99 IP = 1,MAXCHN
0077           BUF(XI+1,IP)=255
0078           BUF(XI+2,IP)=255
0079     99    BUF(XA+1,IP)=255
```

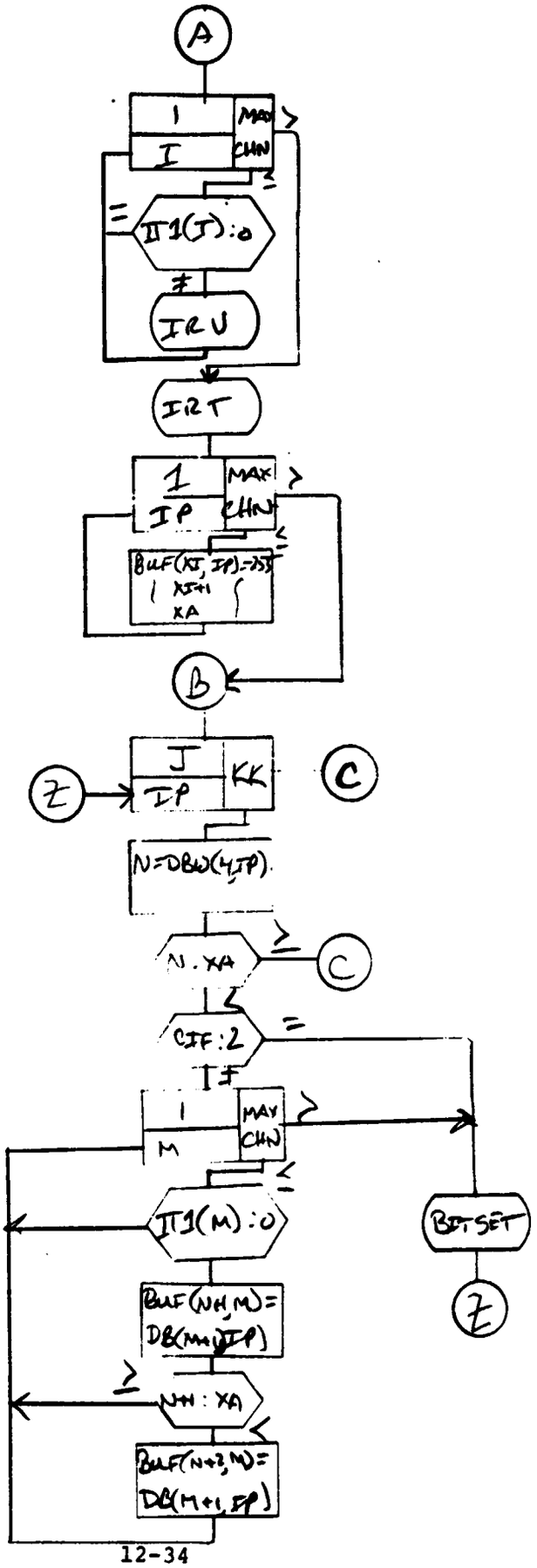
ORIGINAL PAGE IS
OF POOR QUALITY

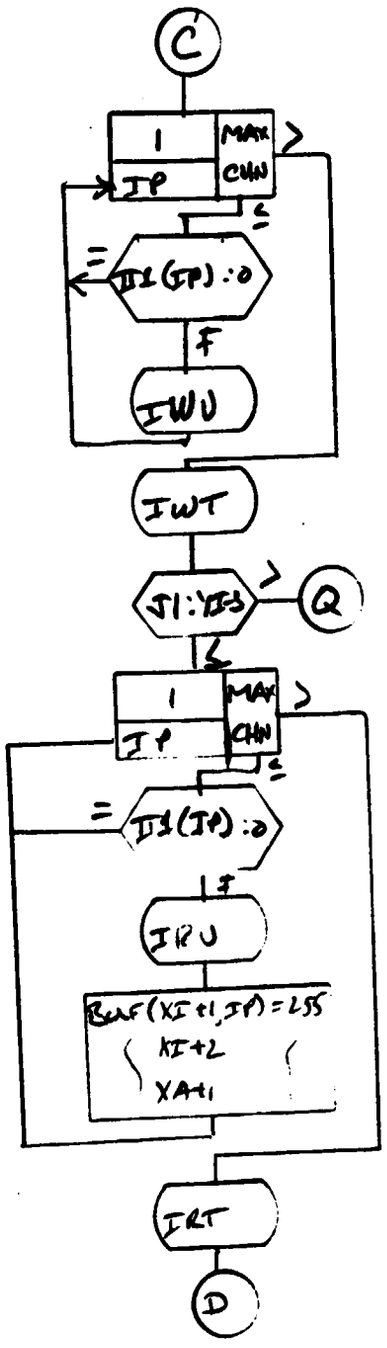
```
0080     2      DO 12 IP = J,KK
0081              N=DBW(4,IP)
0082              IF(N.GE.XA)GO TO 12
0083            IE(CIF.EQ.2) GO TO 10
0084            DO 11 M = 1,MAXCHN
0085            IF(II1(M).EQ.0) GO TO 11
0086            BUF(N+1,M)=DB(M+1,IP)
0087              IF(N+1.GE.XA) GO TO 11
0088            BUF(N+2,M)=DB(M+1,IP)
0089     11     CONTINUE
0090     10     CALL BITSET(N,THB,1)
0091            CALL BITSET(N+1,TH2,1)
0092     12     CONTINUE
0093            DO 13 IP = 1,MAXCHN
0094     13     IF(II1(IP).NE.0) CALL IWV(IP,J1,BUF(1,IP))
0095            CALL IWT(TH,J1,1,THB)
0096            IF(J1.GT.YI-3) GO TO 19
0097            CALL WAIT
0098            DO 14 IP = 1,MAXCHN
0099            IF(II1(IP).EQ.0) GO TO 14
0100            CALL IRV(IP,J1+1,BUF(1,IP))
0101     14     CONTINUE
0102            CALL IRT(TH,J1+1,1,THB)
0103            CALL WAIT
0104            DO 114 IP = 1,MAXCHN
0105            BUF(XI+1,IP)=255
0106            BUF(YI+2,IP)=255
0107            BUF(XA+1,IP)=255
0108     114    CONTINUE
0109            DO 15 IP = J,KK
0110            N = DBW(4,IP)
0111              IF(N.GE.XA)GO TO 15
0112            IF(CIF.EQ.2) GO TO 16
0113            DO 17 M = 1,MAXCHN
0114            IF(II1(M).EQ.0) GO TO 17
0115            BUF(N+1,M)=DB(M+1,IP)
0116              IF(N+1.GE.XA)GO TO 17
0117            BUF(N+2,M)=DB(M+1,IP)
0118     17     CONTINUE
0119     16     CALL BITSET(N,THB,1)
0120            CALL BITSET(N+1,TH2,1)
0121     15       CONTINUE
0122            IF(L.LT.YI-2) GO TO 20
0123            DO 18 IP = 1,MAXCHN
0124     18     IF(II1(IP).NE.0)CALL IWV(IP,L,BUF(1,IP))
0125            CALL IWT(TH,L,1,THB)
0126            CALL WAIT
0127     19       IN = 2
0128     20       L = J1 + 1
0129            RETURN
0130            END
```

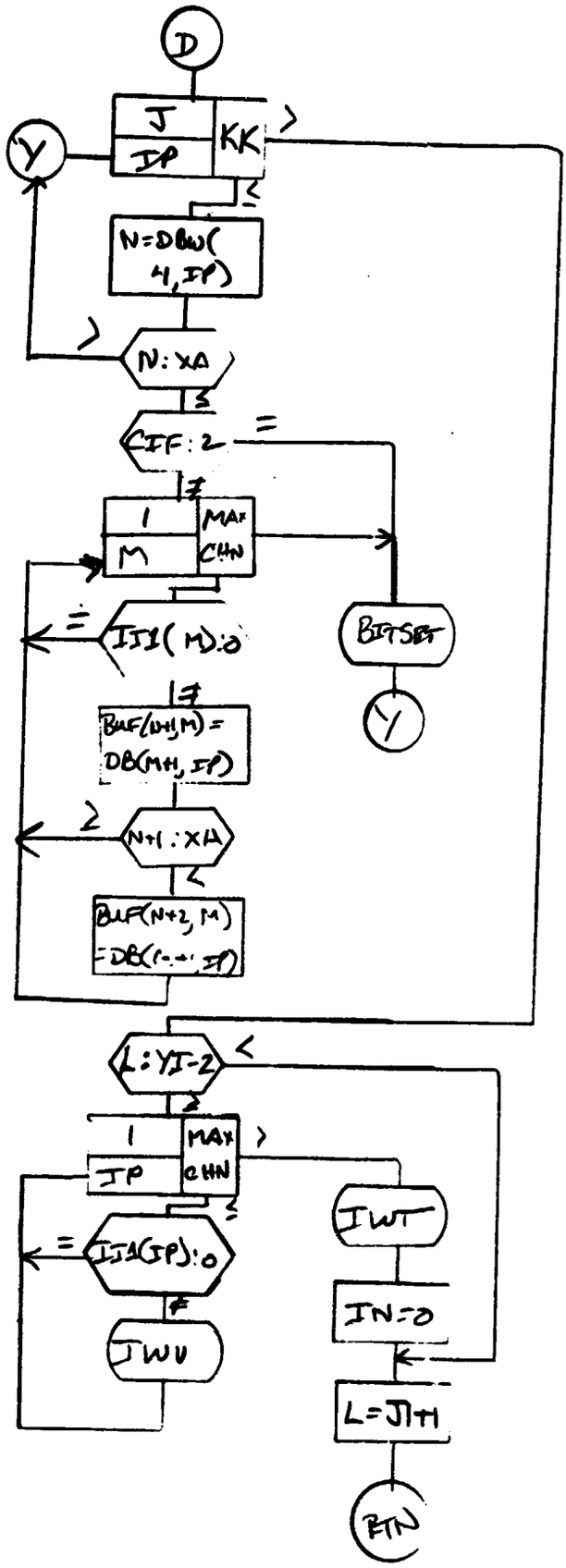
12.3 SUBROUTINE PLOT

ORIGINAL PAGE IS
OF POOR QUALITY




A
I
MAX
CHN
IT1(J):0
IRU
IRT
1
IP
BUF(XJ, IP):=XS
XJ+1
XA
B
J
IP
KK
Z
C
N=DBW(4,IP)
N . XA
CIF:2
1
M
MAX
CHN
IT1(M):0
BIT SET
BUF(NH,M)=
DB(M+1,IP)
NH : XA
BUF(NH+2,M)=
DB(M+1,IP)


C
1
MAX
IP
CHN
=
II1(IP):0
F
IWU
IWT
J1:XI3
Q
1
MAX
IP
CHN
=
II1(IP):0
F
IRU
RMF(XI+1,IP)=255
XI+2
XA+1
IRT
D


D
J
IP
KK
Y
N=DBW(
4,IP)
N:XA
CIF:2
1
M
MAX
CHN
IJ1(M):0
BITSET
Y
BUF(N+1,M)=
DB(M+1,IP)
N+1:XA
BUF(N+2,M)
=DB(N+2,IP)
L:YI-2
1
IP
MAX
CHN
IJ1(IP):0
IWT
IN=0
JWV
L=JI+1
RTN

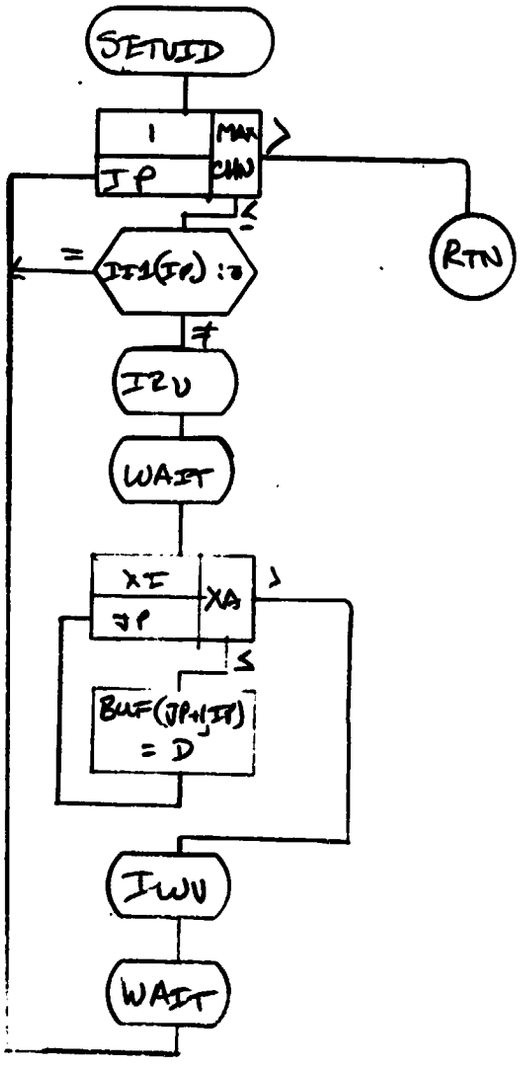
ORIGINAL PAGE IS
OF POOR QUALITY

12.4 SUBROUTINE SETVID

```
HFØRTRAN IV-PLUS VC2-04                14:06:56    31-AUG-77          PAGE 1
SETVID.FTN      /TR:BLØCKS/WR
0001            SUBROUTINE SETVID(LL,D)
0002            INCLUDE 'TØP.INC'
      *  C:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:C
      *  C                                                              C
      *  C                                                              C
      *  C                                                              C
      *  C                                                              C
                C              PLACE A LINE ØF CØNSTANT VALUE ØN THWE CRT      C
                C              T. KELL/LEC/4/77                                  C
                C              SETVID.FTN
0003            INCLUDE 'BØT.INC'
      *  C                                                              C
      *  C                                                              C
      *  C                                                              C
      *  C                                                              C
      *  C:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:CC:C
0004            INCLUDE 'HUGGYBEAR.CØM'
0005  *           INCLUDE '[300,3]CAMSCØMØN.INC'
0006  *         INCLUDE 'SY:[300,3]CAMSPARAM.INC'
0007  *         PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLN=50
      *        1,MAXV=11,NØØTS=209,LLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *        2NØSPWD=6,NØDTWD=10
0008  *         EQUIVALENCE (C1,ACDATE),(C2,ISEG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0009  *         INTEGER C1(469),C2(256),C3(71),C4(348),C5(629)
      *  C*
0010  *         INTEGER ACDATE,SUBCAT,SUPPØR,CATKNT,CATTH
0011  *         BYTE CHNVEC,NØCHAN,NØSUB,DØTCAT,DØTCLU
0012  *         CØMMØN/CØM1/ACDATE(2,MAXACC),CHNVEC(MAXCHN,MAXACC),NØCHAN,NØSUB,
      *        1SUBCAT(MAXSUB),SUBPØP(MAXSUB),CATKNT(MAXCAT),CATTH(MAXCAT),NØDØ,
      *        2NØDU,NØTH,DØTCAT(NDØTS),DØTCLU(NDØTS)
      *  C*
0013  *         INTEGER ADATES,SUNAZ,ANALST,FLDDAY,DØTDAY,PDATE1,TDATE1
0014  *         INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RANDØM,GRID
0015  *         BYTE DELFLG,NØACD,SØILGR,SUNEL,NSTART,NTYPE1,ALP,ALPØ
0016  *         BYTE PCTCT,PCTCTØ,VAR,VARØ,DLABEL,TYPE
0017  *         CØMMØN/CØM2/ISEG,DELFLG,NØACC,ADATES(2,MAXACD),SØILGR(MAXACD),
      *        1SUNEL(MAXACD),SUNAZ(MAXACD),IMDATE(2),ANALST(5),FLDDAY(2),
      *        2DØTDAY(2),NSTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      *        3PDATE3(2),TDATE3(2),NØCAT,CATNAM(MAXCAT),ALP(MAXCAT),ALPØ,
      *        4          PCTCT(MAXCAT),PCTCTØ,VAR(MAXCAT),VARØ
      *  C*
0018  *         INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      *        1UFLAG4
0019  *         INTEGER PFLAG,DSKMNT
0020  *         CØMMØN/CØM3/PFLAG,DSKMNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
      *        1,UFLAG2,UFLAG3,UFLAG4,NEWLAB(MAXSUB)
      *  C*
0021  *         INTEGER TX1,TY1,TX2,TY2,ACDISP,G,B,DTWIND,DØTARY,GMIN,GMAX,FUL
0022  *         INTEGER SPWIND,CLAWND,CLUWND
0023  *         CØMMØN/CØM4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACDISP(2),III(4),G(4),
      *        1R(4),DTWIND(5,NØDTWD),SPWIND(5,NØSPWD),IMWIND(4),NUMDØT,
      *        2DØTARY(NDØTS),GMIN,GMAX,FUL(2,7),CLAWND(8),CLUWND(8)
0024  *         CØMMØN/CØM5/DISKID,RANDØM(NDØTS),GRID(NDØTS),DLABEL(NDØTS),
      *        1TYPE(NDØTS),RECLØC
0025  *           BYTE ACDAC,ACDAI,ACDAJ,ACDAD,BUF,DB,LINE,VAL
0026  *           BYTE PASS,PCH,THR
```

```
SETVID.FTN      /TR:BLØCKS/WR
0027 *          INTEGER CIF,DFLAG,DATE,HMIN,HMAX,I,J,K,LIST,ND,TH,VMIN
0028 *          INTEGER VMAX,XA,XI,YA,YI,TI
0029 *          INTEGER PTR,RECL,WN
0030 *          REAL ØXY,SXY,XYI,XYA
0031 *          COMMON /WUGGY/ ACDAC,ACDAI,ACDAJ,ACDAD,BUF(512,4),CIF,
     *         1 DFLAG,DATE(2),DB(10,NDØTS+1),HMIN,HMAX,I,J,K,LINE(74),
     *         2 LIST(NDØTS),ND,VAL,PASS,ØXY(2),PCH(2),PTR(NDØTS+1),RECL,
     *         3 WN,SXY(2),TH,THB(64),VMIN,VMAX,XA,XI,XYA(2),XYI(2),YA,YI,TI
0032 *          INTEGER DBW(5,NDØTS+1)
0033 *          EQUIVALENCE(DB(1,1),DBW(1,1))
0034 *          BYTE INRAY(36*MAXACC),DØTS(8,MAXACD),DØTP(6)
0035 *          INTEGER CLUS(10,MAXACC),DØTSW(4,MAXACD)
0036 *          REAL CLUS(9,MAXACC)
0037 *          EQUIVALENCE (BUF(1,1),INRAY(1))
0038 *          EQUIVALENCE(INRAY(1),DØTP(1))
0039 *          EQUIVALENCE (DØTP(5),DØTS(1,1))
0040 *          EQUIVALENCE (DØTP(5),DØTSW(1,1))
0041 *          EQUIVALENCE (INRAY(1),CLUS(1,1))
0042 *          EQUIVALENCE (CLUS(1,1),CLUSW(1,1))
0043            BYTE D
0044            INTEGER LL
0045            DØ 1 IP = 1,MAXCHN
0046            IF(III(IP).EQ.0) GØ TØ 1
0047            CALL IRV(IP,LL,BUF(1,IP))
0048            CALL WAIT
0049            DØ 2 JP = XI,XA
0050     2      BUF(JP+1,IP)=D
0051            CALL IWV(IP,LL,BUF(1,IP))
0052            CALL WAIT
0053     1      CØNTINUE
0054            RETURN
0055            END
```

12.4 SUBROUTINE SETVID



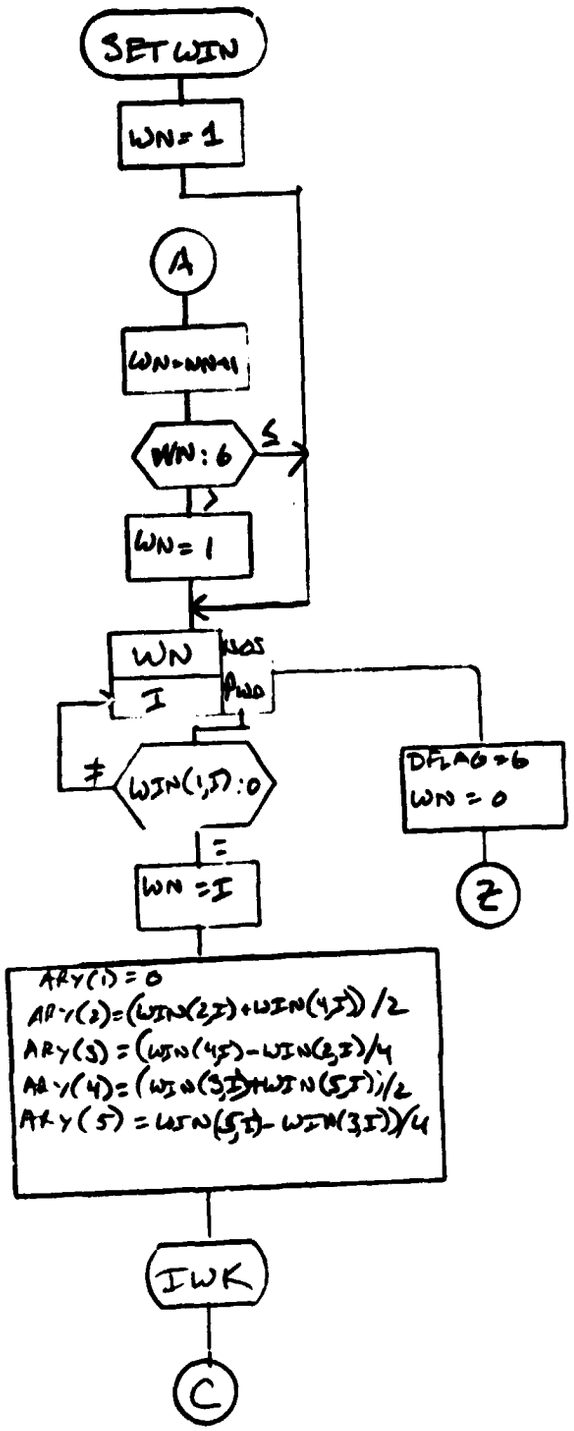
12.5 SUBROUTINE SETWIN

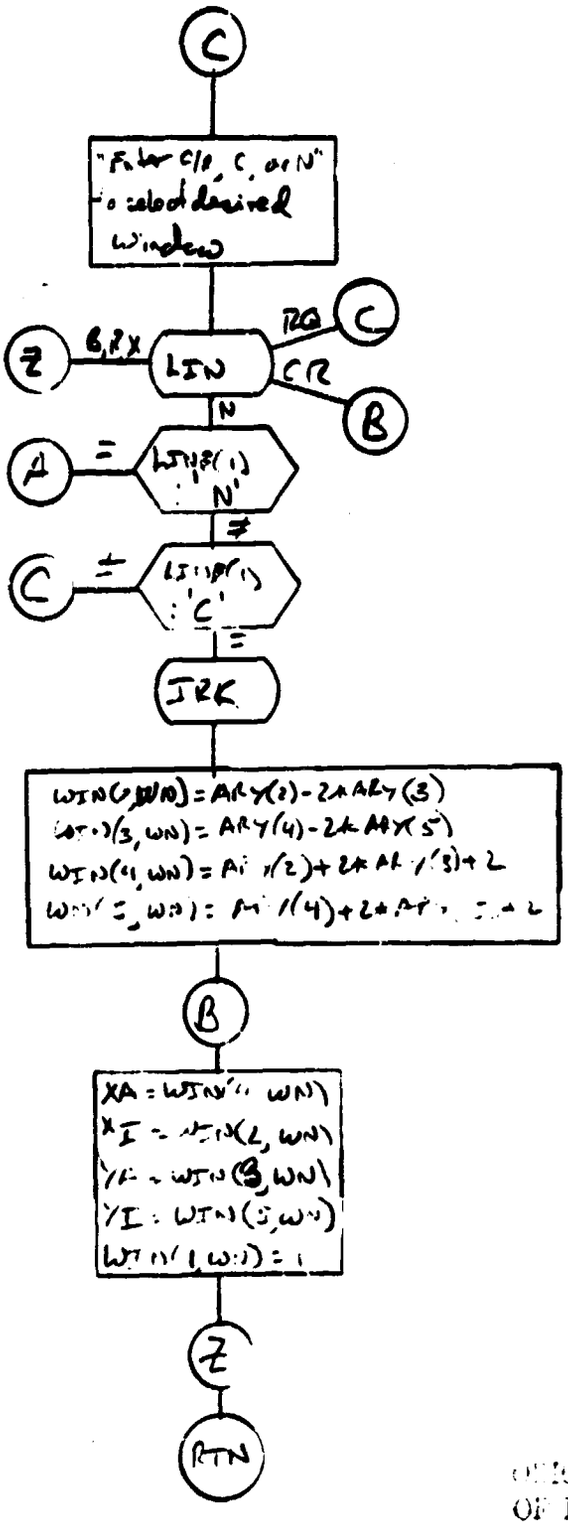
```
HF0RTRAN IV-PLUS V02-04                    14:07:24   31-AUG-77          PAGE 1
SETWIN,FTN       /TR:BL0CKS/WR
0001             SUBR0UTINE SETWIN(I0)
0002             INCLUDE 'T0P.INC'
      *  C!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!C
      *  C                                                                     C
      *  C                                                                     C
      *  C                                                                     C
      *  C                                                                     C
      *  C                                                                     C
         C                SELECT A SCATTERPL0T WIND0W                         C
         C                T. KELL/LEC/4/77                                    C
         C                SETWIN.FTN                                          C
0003             INCLUDE 'B0T.INC'
      *  C                                                                     C
      *  C                                                                     C
      *  C                                                                     C
      *  C                                                                     C
      *  C!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!C
0004               INTEGER ARY(5)
0005               INCLUDE 'HUGGY3EAR.C0M'
0006 *             INCLUDE '[3C0,3]CAMSC0M0N.INC'
0007 *           INCLUDE 'SYI[300,3]CAMSPARAM.INC'
0008 *           PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      *          1,MAXV=11,ND0TS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *          2N0SPWD=6,N0DTW0=10
0009 *           EQUIVALENCE (C1,ACDATE),(C2,ISEG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0010 *           INTEGER C1(469),C2(256),C3(71),C4(348),C5(629)
      *  C*
0011 *           INTEGER ACDATE,SUBCAT,SUBPEP,CATKNT,CATTH
0012 *           BYTE CHNVEC,N0CHAN,N0SUB,D0TCAT,D0TCLU
0013 *           C0MM0N/C0M1/ACDATE(2,MAXACC),CHNVEC(MAXCHN,MAXACC),N0CHAN,N0SUB,
      *          1SUBCAT(MAXSUB),SUBPEP(MAXSUB),CATKNT(MAXCAT),CATTH(MAXCAT),N0D0,
      *          2N0DU,N0TH,D0TCAT(ND0TS),D0TCLU(ND0TS)
      *  C*
0014 *           INTEGER ACDATES,SUNAZ,ANALST,FLDDAY,D0TDAY,PDATE1,TDATE1
0015 *           INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RAND0M,GRID
0016 *           BYTE DELFLG,N0ACC,S0ILGR,SUNEL,NSTART,NTYPE1,ALP,ALP0
0017 *           BYTE PCTCT,PCTCT0,VAR,VAR0,DLABEL,TYPE
0018 *           C0MM0N/C0M2/ISEG,DELFLG,N0ACC,ACDATES(2,MAXACD),S0ILGR(MAXACD),
      *          1SUNEL(MAXACD),SUNAZ(MAXACD),INDATE(2),ANALST(5),FLDDAY(2),
      *          2D0TDAY(2),NSTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      *          3PDATE3(2),TDATE3(2),N0CAT,CATNAM(MAXCAT),ALP(MAXCAT),ALP0,
      *          4          PCTCT(MAXCAT),PCTCT0,VAR(MAXCAT),VAR0
      *  C*
0019 *           INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      *          1UFLAG4
0020 *           INTEGER PFLAG,DSKMNT
0021 *           C0MM0N/C0M3/PFLAG,DSKMNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
      *          1,UFLAG2,UFLAG3,UFLAG4,NEWLAB(MAXSUP)
      *  C*
0022 *           INTEGER TX1,TY1,TX2,TY2,ACDISP,G,B,DTWIND,D0TARY,GMIN,GMAX,FUL
0023 *           INTEGER SPWIND,CLAWND,CLUWND
0024 *           C0MM0N/C0M4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACDISP(2),II1(4),G(4),
      *          1B(4),DTWIND(5,N0DTWD),SPWIND(5,N0SPWD),IMWIND(4),NUMD0T,
      *          2D0TARY(ND0TS),GMIN,GMAX,FUL(2,7),CLAWND(8),CLUWND(8)
0025 *           C0MM0N/C0M5/DISKID,RAND0M(ND0TS),GRID(ND0TS),DLABEL(ND0TS),
      *          1TYPE(ND0TS),RECL0C
0026 *             BYTE ACDAC,ACDAI,ACDAJ,ACDAD,BUF,DB,LINE,VAL
```

```
SETWIN.FTN      /TR:BLOCKS/WR
0027 *          BYTE PASS,PCH,THD
0028 *          INTEGER CIF,DFLAG,DATE,HMIN,HMAX,I,J,K,LIST,ND,TH,VMIN
0029 *          INTEGER VMAX,XA,XI,YA,YI,TI
0030 *          INTEGER PTR,RECL,WN
0031 *          REAL BXY,SXY,XYI,XYA
0032 *          COMMON /HUGGY/ ACDAC,ACDAI,ACDAJ,ACDAD,BUF(512,4),CIF,
     *          1 DFLAG,DATE(2),DB(10,NDBTS+1),HMIN,HMAX,I,J,K,LINE(74),
     *          2 LIST(NDBTS),ND,VAL,PASS,BXY(2),PCH(2),PTR(NDBTS+1),RECL,
     *          3 WN,SXY(2),TH,THB(64),VMIN,VMAX,XA,XI,XYA(2),XYI(2),YA,YI,TI
0033 *          INTEGER DBW(5,NDBTS+1)
0034 *          EQUIVALENCE(DB(1,1),DBW(1,1))
0035 *          BYTE INRAY(36*MAXACC),DBTS(8,MAXACD),DBTP(6)
0036 *          INTEGER CLUSW(18,MAXACC),DBTSW(4,MAXACD)
0037 *          REAL CLUS(9,MAXACC)
0038 *          EQUIVALENCE (BUF(1,1),INRAY(1))
0039 *          EQUIVALENCE(INRAY(1),DBTP(1))
0040 *          EQUIVALENCE (DBTP(5),DBTS(1,1))
0041 *          EQUIVALENCE (DBTP(5),DBTSW(1,1))
0042 *          EQUIVALENCE (INRAY(1),CLUS(1,1))
0043 *          EQUIVALENCE (CLUS(1,1),CLUSW(1,1))
0044            INTEGER WIN(5,NOSPWD)
0045            EQUIVALENCE(WIN(1,1),SPWIND(1,1))
0046            WN = 1
0047     1      DO 2 I = WN,NOSPWD
0048     2      IF(WIN(1,I),EQ,0) GO TO 3
0049            DFLAG = 6
0050     5      WN = 0
0051            RETURN
0052     11     WN = WN + 1
0053            IF(WN.GT.6)WN = 1
0054            GO TO 1
0055     3      WN = I
0056            ARY(1)=0
0057            ARY(2)=(WIN(2,WN)+WIN(4,WN)-2)/2
0058            ARY(3)=(WIN(4,WN)-WIN(2,WN)-2)/4
0059            ARY(4)=(WIN(3,WN)+WIN(5,WN)-2)/2
0060            ARY(5)=(WIN(5,WN)-WIN(3,WN)-2)/4
0061            CALL TWK(ARY)
0062     4      IF(TI.EQ.12)WRITE(10,1000)
0063     1000   FORMAT(' SELECT WINDOW LOCATION FOR SCATTER PLOT:'/
                1 '     ENTER "    CR          " TO ACCEPT WINDOW AS DIS-'/
                2 '     PLAYED, ENTER (N) TO DISPLAY NEXT CURSOR, ENTER (C) TO'/
                4 '$    CURSOR SELECT'
                3 '     DESIRED LOCATION AND SIZE.     >')
0064            CALL LIN(10,LINE,TI,DFLAG)
0065            GO TO (5,5,5,4,6,7)DFLAG
0066     7      IF(LINE(1).EQ.'N') GO TO 11
0067            IF(LINE(1).NE.'C') GO TO 4
0068            CALL IRK(ARY)
0069            WIN(2,WN)=ARY(2)-2*ARY(3)
0070            WIN(3,WN)=ARY(4)-2*ARY(5)
0071            WIN(4,WN)=ARY(2)+ARY(3)*2+2
0072            WIN(5,WN)=ARY(4)+2*ARY(5)+2
0073     6      XA = WIN(4,WN)
0074            XI = WIN(2,WN)
0075            YA = WIN(3,WN)
```

```
SETWIN.FTN      /TR:BLOCKS/WB
0076            YI = WIN(5,WN)
0077            WIN(1,WN)=I1
0078            RETURN
0079            END
```

12.5 SUBROUTINE SETWIN





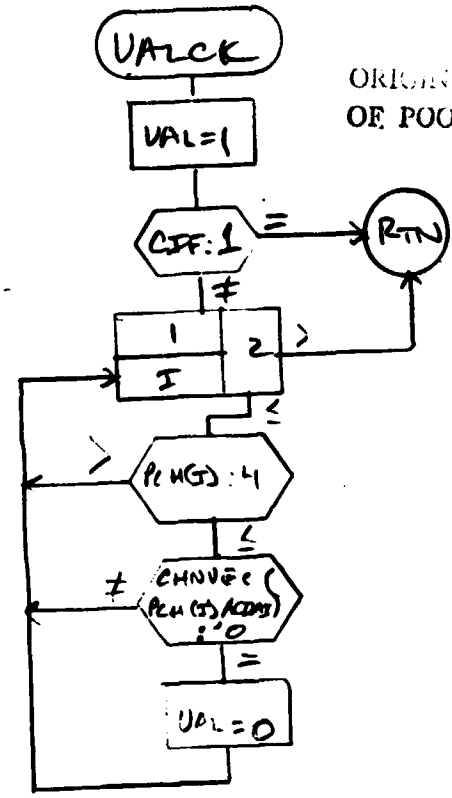
ORIGINAL PAGE IS
OF POOR QUALITY

12.6 SUBROUTINE VALCK

```
HFØRTRAN IV-PLUS V02-04          14:07:51   31-AUG-77          PAGE 1
VALCK.FTN       /TR:RLØCKS/WR
0001            SUBROUTINE VALCK
0002            INCLUDE 'TØP.INC'
      *  C!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!C
      *  C                                                                   C
      *  C                                                                   C
      *  C                                                                   C
      *  C                                                                   C
      *  C                                                                   C
                C                VALIDITY CHECK FØR DGSCPL                  C
                C                T. KELL/LEC/4/77                           C
                C                VALCK.FTN                                  C
0003            INCLUDE 'BØT.INC'
      *  C                                                                   C
      *  C                                                                   C
      *  C                                                                   C
      *  C                                                                   C
      *  C!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!CC!C
0004            INCLUDE 'HUGGYBEAR.CØM'
0005  *         INCLUDE '[300,3]CAMSCØM2Ø.INC'
0006  *       INCLUDE 'SY:[300,3]CAMSPAR2M.INC'
0007  *       PARAMETER MAXCAT=60,MAXSUB=60,MAXCHN=4,NPIX=196,NLIN=117,MAXFLD=50
      *      1,MAXV=11,NDØTS=209,DLSKIP=10,DSSKIP=10,MAXACD=6,MAXACC=4,
      *      2NØSPWD=6,NØDTWD=10
0008  *       EQUIVALENCE (C1,ACDATE),(C2,ISEG),(C3,PFLAG),(C4,TX1),(C5,DISKID)
0009  *       INTEGER C1(449),C2(256),C3(71),C4(348),C5(629)
      *  C*
0010  *       INTEGER ACDATE,SUBCAT,SUBPØP,CATKNT,CATTH
0011  *       BYTE CHNVEC,NØCHAN,NØSUB,DØTCAT,DØTCLU
0012  *       CØMMØN/CØM1/ACDATE(2,MAXACC),CHNVEC(MAXCHN,MAXACC),NØCHAN,NØSUB,
      *      1SUBCAT(MAXSUB),SUBPØP(MAXSUB),CATKNT(MAXCAT),CATTH(MAXCAT),NØDØ,
      *      2NØDU,NATH,DØTCAT(NDØTS),DØTCLU(NDØTS)
      *  C*
0013  *       INTEGER ADATES,SUNAZ,ANALST,FLDDAY,DØTDAY,PDATE1,TDATE1
0014  *       INTEGER PDATE2,TDATE2,PDATE3,TDATE3,CATNAM,DISKID,RANDØM,GRID
0015  *       BYTE DELFLG,NØACQ,SØILGR,SUNEL,ASTART,NTYPE1,ALP,ALPØ
0016  *       BYTE PCTCT,PCTCT2,VAR,VARØ,DLABEL,TYPE
0017  *       CØMMØN/CØM2/ISEG,DELFLG,NØACQ,ADATES(2,MAXACD),SØILGR(MAXACD),
      *      1SUNEL(MAXACD),SUNAZ(MAXACD),IMDATE(2),ANALST(5),FLDDAY(2),
      *      2DØTDAY(2),ASTART,NTYPE1,PDATE1(2),TDATE1(2),PDATE2(2),TDATE2(2),
      *      3PDATE3(2),TDATE3(2),NØCAT,CATNAM(MAXCAT),ALP(MAXCAT),ALPØ,
      *      4          PCTCT(MAXCAT),PCTCT2,VAR(MAXCAT),VARØ
      *  C*
0018  *       INTEGER EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1,UFLAG2,UFLAG3,
      *      1UFLAG4
0019  *       INTEGER PFLAG,DSKMNT
0020  *       CØMMØN/CØM3/PFLAG,DSKMNT,EFLAG1,EFLAG2,EFLAG3,EFLAG4,EFLAG5,UFLAG1
      *      1,UFLAG2,UFLAG3,UFLAG4,NEWLAB(MAXSUB)
      *  C*
0021  *       INTEGER TX1,TY1,TX2,TY2,ACDISP,G,B,DTWIND,DØTARY,GMIN,GMAX,FUL
0022  *       INTEGER SPWIND,CLAWND,CLUWND
0023  *       CØMMØN/CØM4/TX1,TY1,TX2,TY2,IX1,IY1,IX2,IY2,ACDISP(2),I11(4),G(4),
      *      1B(4),DTWIND(5,NØDTWD),SPWIND(5,NØSPWD),IMWIND(4),NUMDØT,
      *      2DØTARY(NDØTS),GMIN,GMAX,FUL(2,7),CLAWND(8),CLUWND(8)
0024  *       CØMMØN/CØM5/DISKID,RANDØM(NDØTS),GRID(NDØTS),DLABEL(NDØTS),
      *      1TYPE(NDØTS),RECL*C
0025  *         BYTE ACDAC,ACDAI,ACDAJ,ACDAD,BUF,DB,LINE,VAL
0026  *         BYTE PASS,PCH,THR
```

```
0027 *          INTEGER CIF,DFLAG,DATE,HMIN,HMAX,I,J,K,LIST,ND,TH,VMIN
0028 *          INTEGER VMAX,XA,XI,YA,YI,TI
0029 *          INTEGER PTR,RECL,WN
0030 *          REAL ØXY,SXY,XYI,XYA
0031 *          COMMON /HUGGY/ ACDAC,ACDAI,ACDAJ,ACDAD,BUF(512,4),CIF,
     *         1 DFLAG,DATE(2),DB(10,ND0TS+1),HMIN,HMAX,I,J,K,LINE(74),
     *         2 LIST(ND0TS),ND,VAL,PASS,ØXY(2),PCH(2),PTR(ND0TS+1),RECL,
     *         3 WN,SXY(2),TH,THB(64),VMIN,VMAX,XA,XI,XYA(2),XYI(2),YA,YI,TI
0032 *          INTEGER DBW(5,ND0TS+1)
0033 *          EQUIVALENCE(DB(1,1),DBW(1,1))
0034 *          BYTE INRAY(36*MAXACC),D0TS(6,MAXACD),D0TP(6)
0035 *          INTEGER CLUSW(18,MAXACC),D0TSW(4,MAXACD)
0036 *          REAL CLUS(9,MAXACC)
0037 *          EQUIVALENCE (BUF(1,1),INRAY(1))
0038 *          EQUIVALENCE(INRAY(1),D0TP(1))
0039 *          EQUIVALENCE (D0TP(5),D0TS(1,1))
0040 *          EQUIVALENCE (D0TP(5),D0TSW(1,1))
0041 *          EQUIVALENCE (INRAY(1),CLUS(1,1))
0042 *          EQUIVALENCE (CLUS(1,1),CLUSW(1,1))
0043             VAL = 1
0044             IF(CIF.EQ.1)RETURN
0045             DØ 1 I = 1,2
0046             IF(PCH(I).GT.4) GØ TØ 1
0047             IF(CHNVEC(PCH(I),ACDAI).EQ.0) VAL = 0
0048     1       CONTINUE
0049             RETURN
0050             END
```

12.6 SUBROUTINE VALCK



ORIGINAL PAGE IS
OF POOR QUALITY

12.7 SUBROUTINE WINDRM

```
HP3000 FORTRAN IV-PLUS V02-04          12:58:32    29-AUG-77          PAGE 1
WINDRM.FTN        /TRIALBOCKS/WR
0001              PROGRAM WINDRM
0002              I = 1
0003              J = 6
0004              CALL CSGDPH(J,I,'WINDOW ERASE',12)
0005              CALL DETACH
0006              CALL ATTACH
0007              CALL WINDER
0008              CALL DETACH
0009              CALL SETEF(53)
0010              END
```

12.7 SUBROUTINE WINDRM

A flow chart of this subroutine can be found in volume 1, page 3-194.

12.8 SUBROUTINE THLOPM

```
MFORTRAN IV-PLUS V02-04          12:58:51    29-AUG-77          PAGE 1
THLOPM.FTN      /TR:BLOCKS/WR
0001            PROGRAM THLOPM
0002            I=1
0003            J=6
0004            CALL CSGOPH(J,I,'THEME LOGICAL OPERATION',23)
0005            CALL DETACH
0006            CALL ATTACH
0007            CALL THMLOP
0008            CALL DETACH
0009            CALL SETEF(53)
0010            END
```

12.8 SUBROUTINE THLOPM

A flow chart of this subroutine can be found in volume 1, page 3-197.

ORIGINAL PAGE IS
OF POOR QUALITY

INDEX TO ALL VOLUMES

The following index lists all computer programs and subroutines found in the text and printouts, and the variables listed in the text. The first number of each description is the volume number, and the remaining number refers to the section in that volume. A preceding L indicates the position of a listing. Therefore, 1-3.3.1 indicates a reference in section 3.3.1 of volume 1, and L2-14.6 indicates that a listing can be found in section 14.6 of volume 2. The list of programs and subroutines is definitive. The list of variables is not complete since those not mentioned in text are not included.

NAME	DESCRIPTION	LINE	COMMENT	OCCURRENCES					LOCATION OF LISTING
A	VARIABLE	1		1-3.5.1.2	1-3.5.1.6	1-3.5.2.6	1-3.5.2.X	3-1.1	
		2		3-9.					
ACAT	VARIABLE	1		1-3.5.2.6					
ACDAT COM1	VARIABLE	1		1-4...9					
ACDATE	VARIABLE	1		1-3.5.1.4	1-3.5.2.5	1-3.5.2.6	1-3.5.2.9		
ACDISP COM4	VARIABLE	1		1-3.4.9	1-3.5.2.3	1-3.5.2.5			
ACLLAP	PROGRAM	1	TASK	1-CONTNTS	1-3.5.2.1	1-3.5.2.6	2-14.		L2-14.
ADATES COM2	VARIABLE	1		1-3.4.9	1-3.5.2.3	1-3.5.2.5	1-3.5.2.6	1-3.5.2.9	
		2		1-3.5.2.X					
ADSK	VARIABLE	1		1-3.5.1.5					
AID	VARIABLE	1		1-3.6.1.5					
AJUL	VARIABLE	1		1-3.5.1.5					
ALABEL	VARIABLE	1		1-3.5.2.6					
ALLUPD PRIVT	SUBROUTINE	1		1-3.5.1.3					L2-3.7
ALP COM2	VARIABLE	1		1-3.4.9					
ALP0 COM2	VARIABLE	1		1-3.4.9					
ALPTAB PRIVT	SUBROUTINE	1							L2-18.4
ALSORT PRIVT	SUBROUTINE	1		1-3.5.2.6					L2-14.4
ANACAT	VARIABLE	1		3-4.4					
ANALYST	VARIABLE	1		1-3.4.9					
ANCL	VARIABLE	1		1-3.5.2.3					
AQLFST PRIVT	SUBROUTINE	1		1-3.5.2.X					L2-21.1
ARAND	VARIABLE	1		1-3.5.2.6					
ARIND	VARIABLE	1		1-3.5.2.6					
ARRY	VARIABLE	1		1-3.5.2.9					
ASEG	VARIABLE	1		1-3.5.1.5					
ASNLUN F4PLIB	SUBROUTINE	1		1-3.5.1.4					
ASSIGN F4PLIB	SUBROUTINE	1		1-3.5.2.6	1-3.5.2.8				
ATTACH F4PLIB	SUBROUTINE	1		1-3.5.2.5	1-3.5.2.6	1-3.5.2.8			
AYR	VARIABLE	1		1-3.5.1.5					
B COM4	VARIABLE	1		1-3.4.9	1-3.5.1.2	1-3.5.1.6	1-3.5.2.3	1-3.5.2.5	
		2		1-3.5.2.X	3-10.	3-19.			
BIASCR	PROGRAM	1	TASK	1-CONTNTS	1-3.5.2.1	1-3.5.2.9	2-18.		L2-18.
BITSET SHARE	SUBROUTINE	1		3-1.					L3-1
BLKTHM SHARE	SUBROUTINE	1		1-3.5.2.5	1-3.5.2.7	1-3.5.2.8	3-2.		L3-2
BLOCK	VARIABLE	1		1-3.5.2.4					
BLOWUP PRVT	SUBROUTINE	1		1-3.5.2.5					L2-13.6
BRFCLU PRVT	SUBROUTINE	1		1-3.5.2.9					L2-14.6
BSTAT OFFICE	PROGRAM	1	TASK	1-CONTNTS	1-3.3.1	1-3.3.5	1-3.5.1.4	2-4.	L2-4.
BUFCLM	VARIABLE	1		1-3.5.2.6					
BUFDOT	VARIABLE	1		1-3.5.2.6					
C	VARIABLE	1		1-3.5.1.2	1-3.5.2.3				
C1	VARIABLE	1		1-3.4.9					
C2	VARIABLE	1		1-3.4.9					
C3	VARIABLE	1		1-3.4.9					
C4	VARIABLE	1		1-3.4.9					
CALP	VARIABLE	1		1-3.5.2.3					
CAMSAVE.INC	FILE	1		1-3.4.9					
CAMSCOMUN.INC	FILE	1		1-3.4.9					
CAMSEX	PROGRAM	1	TASK	1-CONTNTS	1-3.5.2.1	2-7.			L2-7.
CAMSPARAM.INC	FILE	1		1-3.4.6	1-3.4.9	1-3.5.1.3	1-3.5.1.5		
CARD	VARIABLE	1		1-3.5.1.3					

NAME	DESCRIPTION	LINE	COMMENT	OCCURRENCES					LOCATION OF LISTING
CARD.DAT	FILE	1		1-3.5.1.2					
CARDIN PRIVT	FUNCTION	1		1-3.5.1.3					L2-3.10
CAT	VARIABLE	1		1-3.5.1.3					
CATCNT COM1	VARIABLE	1		1-3.4.9	1-3.5.1.4	1-3.5.2.9			
CATLOG PRIVT	FUNCTION	1		1-3.5.1.3					L2-3.2
CATNAM COM2	VARIABLE	1		1-3.4.9	1-3.5.1.3	1-3.5.1.4	1-3.5.2.5	1-3.5.2.6	
		2		1-3.5.2.7	1-3.5.2.9				
CATTH COM1	VARIABLE	1		1-3.4.9	1-3.5.1.4	1-3.5.2.5			
CATTHM PRIVT	SUBROUTINE	1		1-3.5.2.7	2-15.4				L2-15.4
CBRR	VARIABLE	1		1-3.5.2.3					
CDRED PRIVT	SUBROUTINE	1		1-3.5.1.4					L2-4.1
CHAN	VARIABLE	1		1-3.5.2.8					
CHANUC	VARIABLE	1		1-3.5.2.6					
CHL	VARIABLE	1		1-3.5.2.3					
CHNUBC	VARIABLE	1		1-3.5.2.5					
CHNUEC COM1	VARIABLE	1		1-3.4.9	1-3.5.1.4	1-3.5.2.6	1-3.5.2.9		
CHP	VARIABLE	1		1-3.5.2.3					
CLABEL PRIVT	SUBROUTINE	1		1-3.5.2.6	1-3.5.2.7	2-15.7			L2-14.3
		2							L2-15.7
CLACAT	VARIABLE	1		3-4.4					
CLADIS	PROGRAM	1							L2-16.
CLADIS PRIVT	SUBROUTINE	1							L2-16.3
CLASAV PRIVT	SUBROUTINE	1		1-3.5.2.X					L2-21.4
CLASS	SUBROUTINE	1	TASK	1-CONTNTS	1-3.5.2.8				L2-5.3
CLASSMAP.TMP	FILE	1		1-3.4.3	1-3.5.2.2	1-3.5.2.8	1-3.5.2.X		
CLATHM	SUBROUTINE	1		1-3.5.2.8					L2-16.5
CLAWND COM4	VARIABLE	1		1-3.4.9	1-3.5.2.5				
CLLAB	VARIABLE	1		1-3.5.2.6					
CLOSE F4PLIB	SUBROUTINE	1		1-3.5.1.4	1-3.5.2.5	1-3.5.2.6	1-3.5.2.8		
CLREF F4PLIB	SUBROUTINE	1		1-3.5.2.1					
CLSREP PRIVT	SUBROUTINE	1		2-19.					L2-19.
CLUARY	VARIABLE	1		3-4.2					
CLUDIS	PROGRAM	1	TASK	1-CONTNTS	1-3.5.2.1	1-3.5.2.7	2-15.		L2-15.
CLURPT	PROGRAM	1	TASK	1-CONTNTS	1-3.5.2.1	1-3.5.2.9	2-19.		L2-19.
CLUSEL	SUBROUTINE	1		1-3.5.2.5					L3-4.2
CLUSNN PRVT	SUBROUTINE	1		1-3.5.2.9					L2-14.8
CLUSTATS.TMP	FILE	1		1-3.4.5	1-3.5.2.2	1-3.5.2.6	1-3.5.2.9	1-3.5.2.X	
CLUSTATP.TMP	FILE	1		1-3.5.2.5					
CLUSTERMP.TMP	FILE	1		1-3.4.3	1-3.5.2.2	1-3.5.2.7	1-3.5.2.8	1-3.5.2.X	
CLUSTR PRVT	SUBROUTINE	1							L2-5.4
CLUTHM PRIVT	SUBROUTINE	1		1-3.5.2.7	2-15.5				L2-15.5
CLUWND COM4	VARIABLE	1		1-3.4.9	1-3.5.2.5	1-3.5.2.7			
CMASK	VARIABLE	1		1-3.5.2.7					
CNTRL PRIVT	SUBROUTINE	1		1-3.5.1.2	2-2.2				L2-2.2
COM1 GLOBAL	COMMON	1		1-CONTNTS	1-3.4.5	1-3.4.9	1-3.5.1.4	1-3.5.2.2	1-F3-3B
		2		1-3.5.2.7	1-3.5.2.9	1-3.5.2.X			
COM2 GLOBAL	COMMON	1		1-CONTNTS	1-3.3.1	1-3.4.4	1-3.4.9	1-3.5.2.2	1-F3-3D
COM3 GLOBAL	COMMON	1		1-CONTNTS	1-3.4.9	1-3.5.2.2	1-3.5.2.X		1-F3-3E
COM4 GLOBAL	COMMON	1		1-CONTNTS	1-3.4.9	1-3.5.2.2	1-3.5.2.9		1-F3-3F
		2		1-3.5.2.7	1-3.5.2.9	1-3.5.2.X			
COM5 GLOBAL	COMMON	1		1-CONTNTS	1-3.4.9	1-3.5.2.2	1-3.5.2.9	1-3.5.2.X	1-F3-3G
COMLUT PRIVT	SUBROUTINE	1		1-3.5.2.3					L2-9.2
COMPAR PRIVT	FUNCTION	1		1-3.5.1.3					L2-3.11
CONDIS PRIVT	SUBROUTINE	1		1-3.5.2.7	2-15.12				L215.12

ORIGINAL PAGE IS
OF POOR QUALITY

NAME	DESCRIPTION	LINE	COMMENT	OCCURRENCES					LOCATION OF LISTING
CONDIT	PRIVT SUBROUTINE	1		1-3.5.2.7	2-15.13				L215.13
CRUNCH	PRVT SUBROUTINE	1							L2-9
CSGDPH	SHARE SUBROUTINE	1		1-3.5.2.2	1-3.5.2.3	1-3.5.2.4	1-3.5.2.5	1-3.5.2.9	L3-3.
		2		3-3.					
CURDEF	PRVT SUBROUTINE	1							L2-10.4
CURDEF	VARIABLE	1		1-3.5.2.4					
D	VARIABLE	1		1-3.5.2.5					
DATARD	PRVT SUBROUTINE	1							L2-19.3
DATE	F4PLIBSUBROUTINE	1		1-3.5.1.4	1-3.5.2.5				
DAY	VARIABLE	1		1-3.5.1.3					
DCOORD	PRIVT SUBROUTINE	1		1-3.5.1.2	2-2.6				L2-2.6
DDDOT	PRIVT SUBROUTINE	1							L2-1.12
DDIESF	VARIABLE	1		1-3.5.2.4					
DEFALT	PRIVT SUBROUTINE	1		1-3.5.2.7	2-15.11				L215.11
DEFLT2	PRVT SUBROUTINE	1							L2-16.6
DELEAT	PRIVT SUBROUTINE	1		1-3.5.1					L2-1.
DELFLD	PRIVT SUBROUTINE	1		1-3.5.2.4					L2-10.1
DELFLG	COM2 VARIABLE	1		1-3.4.9	1-3.5.2.X				
DETACH	IMALIBSUBROUTINE	1		1-3.5.2.5	1-3.5.2.6	1-3.5.2.8			
DFLAG	VARIABLE	1		1-3.5.2.6					
DGSCPL	SUBROUTINE	1		1-CONTNTS	1-3.5.2.5	2-12.1			L2-12.1
DIRCAT	VARIABLE	1		1-3.5.2.8					
DIRCRE	PRIVT SUBROUTINE	1		1-3.5.1.1					L2-1.
DIRCRE	VARIABLE	1		1-3.5.1.1					
DIRFIL	VARIABLE	1		1-3.5.1.1					
DIRFILE.DAT	FILE	1		1-3.3	1-3.3.1	1-3.5.1.2	1-3.5.1.4	1-3.5.1.5	
		2		1-3.5.1.6	1-3.5.2.2	1-3.5.2.X			
DIRLOD	PRIVT SUBROUTINE	1		1-3.5.1.3					L2-3.3
DIRUPD	IMAUPDPROGRAM	1		1-3.5.1.1					L2-1.
DIRUPD	PRIVT SUBROUTINE	1		1-3.5.1.3	1-3.5.1.5				L2-5.1
DIRUPD	VARIABLE	1		1-3.5.1.1					
DISKID	COM5 VARIABLE	1		1-3.4.9	1-3.5.1.4				
DISKIM	VARIABLE	1		1-3.5.1.3	1-3.5.1.5				
DIST	VARIABLE	1		1-3.4.6					
DK	VARIABLE	1		3-4.2					
DLAB	VARIABLE	1		1-3.4.6					
DLABEL	COM5 VARIABLE	1		1-3.4.4	1-3.4.9	1-3.5.2.4	1-3.5.2.5	1-3.5.2.6	
		2		1-3.5.2.9	1-3.5.2.X	3-11.			
DLABEL	VARIABLE	1		1-3.5.2.4					
DLABEL	COM5 VARIABLE	1		1-3.5.2.6	1-3.5.2.X				
DLSKIP	VARIABLE	1		1-3.5.1.3					
DO1	VARIABLE	1		1-3.5.2.9					
DO2	VARIABLE	1		1-3.5.2.9					
DO3	VARIABLE	1		1-3.5.2.9					
DODU	VARIABLE	1		1-3.5.2.4					
DOI	VARIABLE	1		1-3.5.2.3					
DOLABEL	VARIABLE	1		1-3.5.2.4					
DOPCT	VARIABLE	1		1-3.5.2.9					
DOTARY	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.5				
DOTDAT	COM1 VARIABLE	1		1-3.4.9					
DOTCLU	COM1 VARIABLE	1		1-3.4.9	1-3.5.2.9				
DOTDAT	VARIABLE	1		1-3.5.2.6					
DOTDAY	COM2 VARIABLE	1		1-3.4.9	1-3.5.2.X				

NAME	DESCRIPTION	LINE	COMMENT	OCCURRENCES					LOCATION OF LISTING
DOTFLG	VARIABLE	1		1-3.5.1.3					
DOTGXY.TMP	FILE	1		1-3.4.7	1-3.5.2.2	1-3.5.2.5			
DOTIN	SUBROUTINE	1		1-3.5.2.5	1-3.5.2.9				L3-4.1
DOTLAB	SUBROUTINE	1		1-3.5.2.5					L2-13.4
DOTN	VARIABLE	1		1-3.4.6					
DOTOFF PRIVT	SUBROUTINE	1	*UNLDOT						
DOTOVR	PROGRAM	1	TASK	1-CONTNTS	1-3.5.2.1	1-3.5.2.5	2-11.		L2-11.
DOTPRO	PROGRAM	1	TASK	1-CONTNTS	1-3.5.2.1	1-3.5.2.5			L2-13.
DOTRPT	PROGRAM	1	TASK	1-CONTNTS	1-3.5.2.1	2-17.			L2-17.
DOTSAV PRIVT	SUBROUTINE	1		1-3.5.2.X					L2-21.2
DOTSC_	SUBROUTINE	1		1-3.5.2.9	3-4.4				L3-4.4
DOTS.TMP	FILE	1		1-3.3	1-3.4.4	1-3.5.2.2	1-3.5.2.5	1-3.5.2.6	
		2		1-3.5.2.X					
DOTUPD OFFICE	PROGRAM	1	TASK	1-CONTNTS	1-3.3.1	1-3.3.3	1-3.5.1.3	2-3.	L2-3.
DOTYPE	VARIABLE	1		3-4.4					
DSET SHARE	SUBROUTINE	1		1-3.5.1.3	1-3.5.1.5	1-3.5.2.3	3-5.		L3-5.
DSKCHK SHARE	SUBROUTINE	1		1-3.3.1	1-3.5.1.2	1-3.5.1.3	1-3.5.1.4	1-3.5.1.5	L3-6.
		2		1-3.5.1.6	1-3.5.2.2	3-6.			
DSKID	VARIABLE	1		3-6.					
DSKMNT COM3	VARIABLE	1		1-3.4.9					
DSKTBL	VARIABLE	1		1-3.5.1.1					
DSKTBL..DAT	FILE	1		1-3.3	1-3.3.1	1-3.5.1.6	1-3.5.1.2	1-3.5.2.X	
DSSKIP	VARIABLE	1		1-3.5.1.3					
DTCL10 SHARE	SUBROUTINE	1		3-4.					L3-4.0
DTERM OFFICE	PROGRAM	1	TASK	1-CONTNTS	1-3.3.1	1-3.3.5			L2-5.
DTRM	PROGRAM	1	*DTERM	1-3.5.1.5	2-5				
DTWIND COM4	VARIABLE	1		1-3.4.9	1-3.5.2.4				
DUPCT	VARIABLE	1		1-3.5.2.9					
DUSET	SUBROUTINE	1	*DSET	1-3.5.2.3					
DY	VARIABLE	1		1-3.5.1.2	1-3.5.2.X				
E	VARIABLE	1		3-4.3					
EFLAG1 COM3	VARIABLE	1		1-3.4.3	1-3.4.9	1-3.5.2.7	1-3.5.2.X		
EFLAG2 COM3	VARIABLE	1		1-3.4.9	1-3.5.2.X				
EFLAG3 COM3	VARIABLE	1		1-3.4.9	1-3.5.2.4	1-3.5.2.X			
EFLAG4	VARIABLE	1		1-3.4.6	1-3.4.9	1-3.5.2.6	1-3.5.2.7	1-3.5.2.X	
EFLAG5 COM3	VARIABLE	1		1-3.4.5	1-3.4.9	1-3.5.2.X			
EFWARN PRIVT	SUBROUTINE	1		1-3.5.2.7	2-15.1				L2-15.1
ELAPSE SHARE	SUBROUTINE	1		1-3.5.1.2	1-3.5.1.3	1-3.1.5.5	1-3.5.1.6	1-3.5.2.1	L3-6.1+
		2		1-3.5.2.2	1-3.5.2.5	1-3.5.2.6	1-3.5.2.8	1-3.5.2.9	L3-7.
		3		1-3.5.2.X	3-7.				
EOF	VARIABLE	1		1-3.5.1.5					
ERRMES SHARE	SUBROUTINE	1		1-3.5.1.3	1-3.5.1.5	3-8.			L3-8
ERROR	VARIABLE	1		1-3.5.1.5					
EXFIL	VARIABLE	1		1-3.5.2.5					
EXIT F4PLIB	SUBROUTINE	1		1-3.5.2.5					
EXPTD PRIVT	SUBROUTINE	1		1-3.5.1.2	2-2.3				L2-2.3
F	VARIABLE	1		3-4.3					
F11	VARIABLE	1		1-3.5.2.3	3-5.				
FADE PRIVT	SUBROUTINE	1	*FMAINT	1-3.5.1.5					
FAKCU3 PRIVT	SUBROUTINE	1							L2-10
FDLINT PRIVT	SUBROUTINE	1							L2-16.1
FFFPI SHARE	SUBROUTINE	1		1-3.5.2.3	3-9.				L3-9
FFUNC SHARE	SUBROUTINE	1		1-3.5.2.5	3-10.				L3-10.
FIELD PRIVT	SUBROUTINE	1		1-3.5.1.2	2-2.5				L2-2.5

NAME		DESCRIPTION	LINE	COMMENT	OCCURRENCES					LOCATION OF LISTING
FIELD		VARIABLE	1		1-3.5.2.4	1-3.5.2.8				
FIELDS.TMP		FILE	1		1-3.3.4	1-3.5.2.2	1-3.5.2.4	1-3.5.2.8	1-3.5.2.9	
			2		1-3.5.2.X					
FILE		VARIABLE	1		1-3.5.1.4	3-5.	1-3.5.2.3			
FILE		VARIABLE	1		1-3.5.2.3					
FILEIN		VARIABLE	1		1-3.5.2.2					
FILEOUT		VARIABLE	1		1-3.5.2.2					
FILEST	PRIVT	SUBROUTINE	1		1-3.5.1.1					L2-1.
FILNUM		VARIABLE	1		1-3.5.1.5					
FILTYP		VARIABLE	1		1-3.5.2.X					
FINDOT	PRVT	SUBROUTINE	1		1-3.5.2.5					L2-13.2
FL		VARIABLE	1		1-3.5.2.8	3-11.2				
FLABEL		VARIABLE	1		1-3.5.2.6					
FLAG		VARIABLE	1		1-3.5.2.2	3-4.4	3-6.			
FLDDAY	COM2	VARIABLE	1		1-3.4.9	1-3.5.2.X				
FLDDEF		PROGRAM	1	TASK	1-CONTNTS	1-3.5.2.1	2-10.			L2-10.
FLDEND	PRIVT	SUBROUTINE	1		1-3.5.1.2	2-2.7				L2-2.7
FLDINT		SUBROUTINE	1		1-3.5.1.2	1-3.5.2.8	2-2.17	3-11.		L3-11.+
			2							L2-2.17
FLDLAB		VARIABLE	1		3-11.1					
FLDNAM	PRIVT	SUBROUTINE	1							L2-10.
FLDNAM		VARIABLE	1		1-3.5.2.8					
FLDOFF	PRIVT	SUBROUTINE	1		1-3.5.2.X					L2-21.8
FLDRPT		PROGRAM	1	TASK	1-CONTNTS	1-3.5.2.1	2-20.			L2-20.
FLDRPT	PRIVT	SUBROUTINE	1		1-3.5.2.4	1-3.5.2.9				L2-10.2
FLDSAV	PRIVT	SUBROUTINE	1		1-3.5.2.X					L2-21.3
FLDST	PRIVT	SUBROUTINE	1		1-3.5.1.2	2-2.4				L2-2.4
FLDUPD	OFFICE	PROGRAM	1	TASK	1-CONTNTS	1-3.3.1	1-3.3.3	1-3.3.4	1-3.5.1.2	L2-2.1
			2		2-2.1					
FLGDOT	SHARE	SUBROUTINE	1		1-3.5.1.2	1-3.5.2.4	2-2.18	3-11 1		L3-11.1
			2							+L22.18
FLL		VARIABLE	1		1-3.5.2.8					
FLN		VARIABLE	1		1-3.5.2.3					
FMAINT	PRIVT	SUBROUTINE	1		1-3.5.1.5					L2-5.5
FN		VARIABLE	1		3-5.					
FORM		VARIABLE	1		1-3.5.2.3					
FORDDI.DAT		FILE	1	SYSTEM	1-3.5.1.3					
FPTR		VARIABLE	1		1-3.5.2.4					
FRONT	I-100	SUBROUTINE	1		1-3.5.2.5	1-3.5.2.6	1-3.5.2.8	1-3.5.2.9		
FSTVID	SHARE	SUBROUTINE	1		1-3.5.1.3	1-3.5.1.5	3-12.			L3-12.
FTRNFR	PRIVT	SUBROUTINE	1		1-3.5.2.2					L2-8.1
FUL	COM4	VARIABLE	1		1-3.4.9	1-3.5.2.5				
FULL		VARIABLE	1		1-3.5.2.5					
FUL012	PRIVT	SUBROUTINE	1	*FUL014						L2-9.5
FUL013		PROGRAM	1	TASK	1-CONTNTS	1-3.5.2.1	1-3.5.2.3	2-9.		L2-9.
FUL014	PRIVT	SUBROUTINE	1	TASK	1-3.5.2.3					L2-9.4
FUCLOS	I-100	SUBROUTINE	1		3-12.					
FUDLTE	I-100	SUBROUTINE	1		3-12.					
FUDSET	I-100	SUBROUTINE	1		3-12.					
FUOPEN	I-100	SUBROUTINE	1		3-12.					
FUREAD	I-100	SUBROUTINE	1		3-12.					
FURWND	I-100	SUBROUTINE	1		3-12.					
FUWAIT	I-100	SUBROUTINE	1		3-12.					
FUWRIT	I-100	SUBROUTINE	1		3-12.					

NAME	DESCRIPTION	LINE	COMMENT	OCCURRENCES					LOCATION OF LISTING
G	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.3	1-3.5.2.5			
GABI	PRIVT SUBROUTINE	1		1-3.5.2.3					L2-9.1
GB	VARIABLE	1		1-3.5.2.3					
GBCALC	PRIVT SUBROUTINE	1							L2-9.6
GETADR	F4PLIBSUBROUTINE	1		1-3.5.1.4					
GETC00	SHARE SUBROUTINE	1		1-3.5.2.7	2-15.8	3-13.			L3-11.+
		2							L2-15.8
GLOBAL.TMP	FILE	1		1-3.4.9	1-3.5.2.2	1-3.5.2.5	1-3.5.2.9		
GMAX	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.5				
GMIN	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.5				
GRENO	VARIABLE	1		1-3.5.2.9					
GRENOB	VARIABLE	1		1-3.5.2.5					
GRENSS	VARIABLE	1		1-3.5.2.9					
GRID	COM5 VARIABLE	1		1-3.4.9					
GTYPE	PRVT SUBROUTINE	1		1-3.5.2.5					L2-13.7
HDRREC.DAT	FILE	1		1-3.2.2					
HEXD	PRIVT SUBROUTINE	1							L2-4.3
HEADIN	PRIVT SUBROUTINE	1		1-3.5.1.5					L2-5.9
HFG	VARIABLE	1		1-3.5.2.3					
HFL	VARIABLE	1		1-3.5.2.3					
HDCUTT	PRIVT SUBROUTINE	1							L2-9.4
HPROS	SHARE SUBROUTINE	1		1-3.5.1.5	3-14.				L3-14.
HRBAD	SUBROUTINE	1	*DSET	1-3.5.2.3					
HSEKPG	PRIVT SUBROUTINE	1		1-3.5.2.5					L2-13.5
		2							L2-19.4
HSIZ	VARIABLE	1		1-3.5.2.3					
HUFY	SHARE SUBROUTINE	1		1-3.5.1.5	3-15.				L3-15.
I	VARIABLE	1		1-3.5.1.2	1-3.5.1.6	1-3.5.2.5	1-3.5.2.X	3-1.1	
		2		3-3.1	3-17.				
IBUF	VARIABLE	1		1-3.5.1.2	1-3.5.1.4	1-3.5.2.5	3-2.		
IBYTE	IMALIBSUBROUTINE	1		1-3.5.2.5					
IC	VARIABLE	1		1-3.5.2.7	3-13.				
ICOUNT	VARIABLE	1		1-3.5.2.8					
ID	VARIABLE	1		1-3.5.1.2	1-3.5.1.5				
IDATE	F4PLIBSUBROUTINE	1		1-3.5.2.5	1-3.5.2.9				
IDENT1	PRIVT SUBROUTINE	1		1-3.5.2.8					
IER	VARIABLE	1		1-3.4.6					
IERR	VARIABLE	1		1-3.5.1.?					
IFST	VARIABLE	1		1-3.5.2.4					
II	VARIABLE	1		1-3.5.2.5	1-3.5.2.9	3-7.			
III	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.3	1-3.5.2.5			
IIX1	VARIABLE	1		1-3.5.2.3					
IIX2	VARIABLE	1		1-3.5.2.3					
IIY1	VARIABLE	1		1-3.5.2.3					
IIY2	VARIABLE	1		1-3.5.2.3					
ILABEL	VARIABLE	1		1-3.5.2.6					
IMAUPD	UFFICE PROGRAM	1	TASK	1-3.3	1-3.3.1	1-3.3.2	1-3.3.3	1-3.5.1	L2-1.
IMDATE	COM2 VARIABLE	1		1-3.4.9	1-3.5.2.X				
IMWIND	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.5				
INCLDS	PRIVT SUBROUTINE	1		1-3.5.2.9					L2-19.2
INOBRT	PRIVT SUBROUTINE	1		1-3.5.1.2					L2-2.2
INIT	INTRAC PROGRAM	1	TASK	1-CONTNTS	1-3.3.1	1-3.3.3	1-3.3.4	1-3.3.5	L2-8.
		2		1-3.5.2.1	1-3.5.2.2	2-8			
INIT	VARIABLE	1		1-3.5.1.3					

NAME	DESCRIPTION	LINE	COMMENT	OCCURRENCES					LOCATION OF LISTING
INTFF	F4PLIB SUBROUTINE	1		1-3.5.2.5	1-3.5.2.6	1-3.5.2.8	1-3.5.2.9		
INTLZE	PRIVT SUBROUTINE	1		1-3.5.2.2					L2-8.2
INX	VARIABLE	1		1-3.5.1.2					
IO	VARIABLE	1		1-3.5.2.4	1-3.5.2.5	1-3.5.2.9	3-3.	3-4.1	
IOP	VARIABLE	1		1-3.5.2.5	3-2.				
IOPT	VARIABLE	1		1-3.5.1.2					
IP	VARIABLE	1		1-3.5.2.3					
IRDDEF.TMP	FILE	1		1-3.5.2.4					
IRESUB	VARIABLE	1		1-3.5.2.8					
IRREG3	SUBROUTINE	1	TASK	1-3.5.2.4					
IRT	IMALIB SUBROUTINE	1		1-3.5.2.5					
IRV	IMALIB SUBROUTINE	1		1-3.5.2.5					
IOPRNT	PRIVT SUBROUTINE	1							L2-4.2
		2		3-4.2					
ISEG	COM2 VARIABLE	1		1-3.4.9	1-3.5.2.3	1-3.5.2.6	1-3.5.2.9		
ISET	VARIABLE	1		3-13.					
ITUCHN	VARIABLE	1		1-3.5.2.6					
IU	VARIABLE	1		3-17.					
IWL	IMALIB SUBROUTINE	1		1-3.5.2.5					
IWT	IMALIB SUBROUTINE	1		1-3.5.2.5					
IX	VARIABLE	1		1-3.5.2.3	1-3.5.2.7	3-4.2			
IX1	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.3	1-3.5.2.7			
IX2	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.3	1-3.5.2.7			
IXY	VARIABLE	1		1-3.5.1.2					
IY	VARIABLE	1		1-3.5.2.3	1-3.5.2.7				
IY1	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.3	1-3.5.2.7			
IY2	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.3	1-3.5.2.7			
J	VARIABLE	1		1-3.5.1.2	1-3.5.1.6	1-3.5.2.X			
JDIR	PROGRAM	1	=IMAUPD						L2-1.
JJ	VARIABLE	1		3-11.2					
JJ	VARIABLE	1		1-3.5.2.8					
JJL	VARIABLE	1		1-3.5.2.8					
JUL10	VARIABLE	1		1-3.5.1.2	1-3.5.2.X				
JULDAT	VARIABLE	1		1-3.5.1.5					
JULIAN	PRIVT FUNCTION	1							L2-3. +
		2							L2-5.8
JULIAN	PRIVT SUBROUTINE	1		1-3.5.1.2	1-3.5.1.3	1-3.5.2.X	2-2.16		L2-1 +
		2							L2-2.16
K	VARIABLE	1		1-3.5.2.5	1-3.5.2.X				
KAUTH	SUBROUTINE	1		1-3.5.1.4					L2-1.
KK	VARIABLE	1		3-4.1					
KNN	VARIABLE	1		1-3.4.6	1-3.5.2.6				
KNNPRN	PRIVT SUBROUTINE	1		1-3.5.2.6					L2-14.5
KOMBRT	PRIVT SUBROUTINE	1		1-3.5.1.2					L2-2.2
KRW	VARIABLE	1		1-3.5.2.2					
L	VARIABLE	1		3-17.					
LABEL	VARIABLE	1		1-3.5.1.3	1-3.5.2.8				
LABNUM	VARIABLE	1		1-3.5.1.3					
LE	VAEIABLE	1		1-3.5.2.3					
LECTAP	SHARE SUBROUTINE	1		1-3.5.1.3	1-3.5.1.5	3-16.			L3-16.
LGBC	VARIABLE	1		1-3.5.2.3					
LGF	VARIABLE	1		1-3.5.2.3					
LGL	VARIABLE	1		1-3.5.2.3					

ORIGINAL PAGE IS OF POOR QUALITY

NAME	DESCRIPTION	LINE	COMMENT	OCCURRENCES					LOCATION OF LISTING
LIN	SHARE SUBROUTINE	1		1-3.5.2.3	1-3.5.2.4	1-3.5.2.5	3-17.		L3-17.
LIST1	PRIVT SUBROUTINE	1		1-3.5.2.7	2-15.3				L2-15.3
LIST2	PRIVT SUBROUTINE	1		1-3.5.2.7	2-15.15				L2-15.5
LIST3	PRIVT SUBROUTINE	1							L3-16.4
LOOKUP	VARIABLE	1							L2-6.11
LOWHL	VARIABLE	1		1-3.5.2.8					
LREED	SUBROUTINE	1	=DSET	1-3.5.1.5					
LRJUNK	PRIVT SUBROUTINE	1							L2-12.2
LS	VARIABLE	1		1-3.5.2.3					
LUN	VARIABLE	1		1-3.4.6	1-3.5.2.3	3-5.			
LUN	VARIABLE	1		1-3.5.2.3					
LUT	VARIABLE	1		1-3.5.2.3					
M	VARIABLE	1		1-3.5.1.2	1-3.5.1.6	1-3.5.2.X			
MAPUPD	PRIVT SUBROUTINE	1		1-3.5.1.5					L2-5.2
MAXACC	VARIABLE	1		1-3.4.5	1-3.4.9	1-3.5.2.9			
MAXACD	VARIABLE	1		1-3.4.9					
MAXCAT	VARIABLE	1		1-3.5.1.3					
MAXCHN	VARIABLE	1		1-3.4.9	1-3.5.1.3	1-3.5.1.5			
MAXSUB	VARIABLE	1		1-3.4.9					
MB	VARIABLE	1		1-3.5.2.5	3-2.				
MDODU	PROGRAM	1	=FLDUPD						L2-2.1
MENSTD	PRIVT SUBROUTINE	1		1-3.5.2.9					L2-19.1
MFLDS	VARIABLE	1		1-3.5.2.4					
MGL	VARIABLE	1		1-3.5.2.3					
MIXDIS	PRIVT SUBROUTINE	1		1-3.5.2.7	2-15.14				L215.14
MIXED	PRIVT SUBROUTINE	1		1-3.5.2.7	2-15.16				L215.16
ML	VARIABLE	1		1-3.5.2.5	3-2.				
MO	VARIABLE	1		1-3.5.1.2	1-3.5.2.X				
MR	VARIABLE	1		1-3.5.2.2	1-3.5.2.5	3-2.			
MTXTFG	VARIABLE	1		1-3.5.2.3	3-5.				
MU	VARIABLE	1		1-3.5.2.5	3-2.				
MX	VARIABLE	1		1-3.5.2.3	1-3.5.2.7				
MXC	VARIABLE	1		1-3.5.2.3					
MY	VARIABLE	1		1-3.5.2.3	1-3.5.2.7				
N	VARIABLE	1		1-3.5.1.2	1-3.5.1.6	1-3.5.2.5	1-3.5.2.6	1-3.5.2.X	
		2		3-3.	3-9.				
N	VARIABLE	1		1-3.5.2.6	3-3.				
NACQ	VARIABLE	1		1-3.5.2.6					
NBIT	VARIABLE	1		1-3.5.2.3					
NC	VARIABLE	1		1-3.5.2.3					
NCAR	VARIABLE	1		1-3.5.2.3					
NCMAX	VARIABLE	1		1-3.5.2.3					
NCNTRL	PRIVT SUBROUTINE	1		1-3.5.1.2	2-2.20				L2-2.20
NCPR	VARIABLE	1		1-3.5.2.3					
NCR	VARIABLE	1		1-3.5.2.3					
ND	VARIABLE	1		1-3.4.6					
NDOPIX	VARIABLE	1		1-3.5.2.8					
NDOT	VARIABLE	1		1-3.5.1.5					
NDOTS	VARIABLE	1		1-3.4.4	1-3.4.9	1-3.5.1.3			
NDSPR	VARIABLE	1		1-3.5.2.3					
NDUPIX	VARIABLE	1		1-3.5.2.8					

NAME	DESCRIPTION	LINE COMMENT		OCCURRENCES					LOCATION OF LISTING
NEWLAB	COM3	VARIABLE	1	1-3.4.9	1-3.5.2.6	1-3.5.2.7			
NF		VARIABLE	1	1-3.5.2.3					
NFIELD	PRIVT	SUBROUTINE	1	1-3.5.1.2	2-2.21				L2-2.21
NFL		VARIABLE	1	1-3.5.2.4					
NFLD		VARIABLE	1	1-3.5.2.4					
NFLDST	PRIVT	SUBROUTINE	1	1-3.5.1.2	2-2.19				L2-2.19
NLIN		VARIABLE	1	1-3.5.1.3	1-3.5.1.5	1-3.5.2.8			
NLP		VARIABLE	1	1-3.5.2.8					
NN.TMP		FILE	1	I-CONTNTS	1-3.4.6	1-3.5.2.6	1-3.5.2.7		
NNNNYYDDD.DAT		FILE	1	1-3.5.1.1					
NOACQ	COM2	VARIABLE	1	1-3.4.9	1-3.5.2.9	1-3.5.2.X			
NOCAT	COM2	VARIABLE	1	1-3.4.9	1-3.5.1.4	1-3.5.2.5	1-3.5.2.9	1-3.5.2.X	
NOCHAN	COM1	VARIABLE	1	1-3.4.9	1-3.5.1.4				
NODO	COM1	VARIABLE	1	1-3.4.9	1-3.5.1.4	1-3.5.2.9			
NODOT		VARIABLE	1	3-4.4					
NODTND		VARIABLE	1	1-3.4.9					
NODTWD		VARIABLE	1	1-3.4.9					
NODU	COM1	VARIABLE	1	1-3.4.9	1-3.5.1.4	1-3.5.2.9			
NOFLD		VARIABLE	1	3-11.1					
NOLIN		VARIABLE	1	1-3.5.2.8					
NOSPWD		VARIABLE	1	1-3.4.8	1-3.4.9				
NOSUB	COM1	VARIABLE	1	1-3.4.5	1-3.4.6	1-3.4.9	1-3.5.1.4	1-3.5.2.7	
			2	1-3.5.2.9					
NOTH	COM1	VARIABLE	1	1-3.4.9	1-3.5.1.4	1-3.5.2.9			
NPIX		VARIABLE	1	1-3.5.1.3	1-3.5.1.5	1-3.5.2.8			
NPIX4		VARIABLE	1	1-3.5.1.5					
NPTS		VARIABLE	1	3-11.2					
NR		VARIABLE	1	1-3.5.2.2	3-19.				
NRPDS		VARIABLE	1	1-3.5.2.3					
NS		VARIABLE	1	1-3.5.2.3					
NSAMP		VARIABLE	1	1-3.5.2.8	3-11.2				
NSEGND	PRIVT	SUBROUTINE	1	1-3.5.1.2	2-2.22				L2-2.22
NSTART	COM2	VARIABLE	1	1-3.4.9	1-3.5.2.X				
NT		VARIABLE	1	1-3.5.2.5	3-2.				
NTH		VARIABLE	1	1-3.5.2.8					
NTYPE1	COM2	VARIABLE	1	1-3.4.9	1-3.5.2.5	1-3.5.2.6			
NUCAT		VARIABLE	1	1-3.5.2.7					
NUMDOT	COM4	VARIABLE	1	1-3.4.9	1-3.5.2.5				
NU		VARIABLE	1	1-3.5.2.4	1-3.5.2.8	3-11.1			
NX		VARIABLE	1	1-3.5.2.3					
OPEN	F4PLIB	SUBROUTINE	1	1-3.5.1.4	1-3.5.2.9				
OPMESS	PRIVT	SUBROUTINE	1	1-3.5.1.3	1-3.5.1.5				L2-3.13
			2						L2-5.7
OUTFILE.DAT		FILE	1	1-3.5.1.2					
OUTPUT	IMALIB	SUBROUTINE	1	1-3.5.2.5	1-3.5.2.6	1-3.5.2.8	1-3.5.2.9		
P		VARIABLE	1	3-3.	3-19.				
PAINT	PRIVT	SUBROUTINE	1						L2-10.
PCTCT	COM2	VARIABLE	1	1-3.4.9					
PCTCT0	COM2	VARIABLE	1	1-3.4.9					
PDATE1	COM2	VARIABLE	1	1-3.4.9	1-3.5.2.X				
PDATE2	COM2	VARIABLE	1	1-3.4.9	1-3.5.2.X				
PDATE3	COM2	VARIABLE	1	1-3.4.9	1-3.5.2.X				

NAME		DESCRIPTION	LINE	COMMENT	OCCURRENCES					LOCATION OF LISTING
PERDO		VARIABLE	1		1-3.5.2.8					
PERDU		VARIABLE	1		1-3.5.2.8					
PERTH		VARIABLE	1		1-3.5.2.8					
PERUND		VARIABLE	1		1-3.5.2.8					
PFLAG	COM3	VARIABLE	1		1-3.4.3	1-3.4.5	1-3.4.9	1-3.5.2.X		
PLOT		SUBROUTINE	1		1-3.5.2.5					L2-12.3
POLYCR		PROGRAM	1							L2-10.
POLYCR		SUBROUTINE	1							L2-10.3
PRESET	PRIVT	SUBROUTINE	1							L2-1.
PRMUPD	INTRAC	PROGRAM	1	TASK	1-CONTNTS	1-3.3.1	1-3.3.3	1-.3.34	1-3.3.5	L2-21.
			2		1-3.3.2.1	1-3.5.2.X				
PROCED	PRIVT	SUBROUTINE	1							L2-13.3
PTR		VARIABLE	1		1-3.5.1.3	1-3.5.1.5	3-6.			
Q10		SUBROUTINE	1		1-3.5.1.4					L2-1.
R		VARIABLE	1		1-3.5.1.4					
RANDOM	COM5	VARIABLE	1		1-3.4.9	1-3.5.2.6				
RANDOT		VARIABLE	1							L2-3.5
RDCARD	PRIVT	FUNCTION	1		1-3.5.1.3					L2-3.8
RDCLMN	PRIVT	SUBROUTINE	1		1-3.5.2.6					L2-14.2
RDDIR	PRIVT	SUBROUTINE	1		1-3.5.1.2	2-2.9				L2-2.9.
RDDISK	PRIVT	SUBROUTINE	1		1-3.5.1.1					L2-1.
RDDODU	PRIVT	SUBROUTINE	1		1-3.5.1.2	2-2.11				L2-2.10
RDDOT	PRIVT	SUBROUTINE	1		1-3.5.1.2	2-2.10				L2-2.11
RDFLD	PRIVT	SUBROUTINE	1							L2-21.9
RDHEAD	PRIVT	SUBROUTINE	1		1-3.5.1.1					L2-1.
RDODAT		SUBROUTINE	1		1-3.5.2.6					L2-14.1
RDXYD3	PRIVT	SUBROUTINE	1							L2-10.6
RECPCT		VARIABLE	1		1-3.5.2.8					
RECPRN		SUBROUTINE	1		1-3.5.2.8					L2-16.2
RECPRO		PROGRAM	1	TASK	1-CONTNTS	1-3.5.2.8	2-16.			L2-16.
REPORT	PRIVT	SUBROUTINE	1		1-3.5.1.1	1-3.5.1.5	2-15.10			L2-1 +
			2							L2-5.10
			3							L2-14.7
			4							L215.10
REPROP	PRIVT	SUBROUTINE	1	TASK	1-CONTNTS	1-3.5.2.8				L2-16.
REPRTN	PRIVT	SUBROUTINE	1							L2-14.9
REQUES	F4PLIB	SUBROUTINE	1		1-3.5.2.1					
RFAC		VARIABLE	1		1-3.5.2.3					
RIELD		VARIABLE	1		1-3.5.2.8					
RNUM		VARIABLE	1		1-3.5.2.9					
ROFF	PRIVT	SUBROUTINE	1		1-3.5.2.9					L2-18.1
ROFNO		VARIABLE	1		1-3.5.2.9					
RPTGEN	PRIVT	SUBROUTINE	1		1-3.5.2.X					L2-21.7
RREAD	SHARE	SUBROUTINE	1		1-3.5.1.3	1-3.5.1.5	3-18			L3-18.
RS		VARIABLE	1		3-19.					
RSIZ		VARIABLE	1		1-3.5.2.3					
RSKIP		VARIABLE	1		1-3.5.2.3					
RWD		VARIABLE	1		3-5.					
S		VARIABLE	1		3-3.					
SCATXY.TMP		FILE	1		1-3.4.8	1-3.5.2.2	1-3.5.2.5			
SCPLOT	DRIVER	PROGRAM	1	TASK	1-CONTNTS	1-3.5.2.1				L2-12.
SDLINE			1							L2-10.7
SDPNT	PRIVT	SUBROUTINE	1							L2-10.8
SE		VARIABLE	1		1-3.5.2.3					

NAME	DESCRIPTION	LINE	COMMENT	OCCURRENCES					LOCATION OF LISTING
SECNDS	F4PLIBSUBROUTINE	1		1-3.5.1.4	.-3.5.2.5				
SEGDEL	OFFICE PROGRAM	1	TASK	1-CONTNTS	1-3.3.3	1-3.3.4	1-3.3.5	1-3.5.1.6	L2-6.
		2		2-6.					
SEGEND	PRIVT SUBROUTINE	1		1-3.5.1.2	2-2.8				L2-2.8
SEGNO	VARIABLE	1		1-3.5.2.X	3-6.				
SEGNUM	VARIABLE	1		1-3.5.1.5					
SELDOT	PRIVT SUBROUTINE	1		1-3.5.2.9					L2-18.2
SETBIT	IMALIBSUBROUTINE	1		1-3.5.2.5					
SETEF	F4PLIBSUBROUTINE	1		1-3.5.2.5	1-3.5.2.9				
SETVID	SUBROUTINE	1		1-3.5.2.5					L2-12.4
SETWIN	SUBROUTINE	1		1-3.5.2.5					L2-12.5
SHELL	SHARE SUBROUTINE	1		1-3.5.2.5	3-19.				L3-19.
SKIP	PRIVT FUNCTION	1		1-3.5.1.3					L2-3.9
SOILGR	COM2 VARIABLE	1		1-3.4.9	1-3.5.2.5				
SORT	PRIVT SUBROUTINE	1							L2-10.X
SORTRC	PRIVT SUBROUTINE	1							L2-5.6
SPWIND	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.5				
SRDISK	PRIVT SUBROUTINE	1		1-3.5.1.1					L2-1.
SRDISK	VARIABLE	1		1-3.5.1.1					
SS	VARIABLE	1		1-3.5.2.3					
SSSSYYDDD.DAT	FILE	1		1-3.5.1.1					
START	VARIABLE	1		1-3.5.1.3					
STASAV	PRIVT SUBROUTINE	1		1-3.5.2.X					L2-21.5
STATFIL.TMP	FILE	1		1-3.5.2.2					
STRAYS	PRIVT SUBROUTINE	1		1-3.5.1.1					L2-1.
STYPE	PRIVT SUBROUTINE	1		1-3.5.2.5					L2-13.1
SUBCAT	COM1 VARIABLE	1		1-3.4.9	1-3.5.1.4	-13.5.2.	9-		
SUBPOP	COM1 VARIABLE	1		1-3.4.9	1-3.5.1.4				
SUBSTR	SHARE SUBROUTINE	1		1-3.5.1.3	1-3.5.1.5	1-3.5.1.6	1-3.5.2.X	3-20.	L2-3.12
		2							L2-6. +
		3							L2-13.+
		4							L2-21.6
		5							L3-20.
SUD	VARIABLE	1		1-3.5.2.3					
SUNAZ	COM2 VARIABLE	1		1-3.4.9					
SUNEL	COM2 VARIABLE	1		1-3.4.9	1-3.5.2.6				
SWCLR	PRIVT SUBROUTINE	1							L2-10.9
T2	PRIVT SUBROUTINE	1	T2DRMOD						
T2DR	PRIVT SUBROUTINE	1							L2-1.
TABLE	SHARE SUBROUTINE	1		3-4.					L3-4.
TAPSCN	PRIVT SUBROUTINE	1		1-3.5.1.1					L2-1.
TC	VARIABLE	1		1-3.5.2.7	3-13.				
TCHLST	SUBROUTINE	1		1-3.5.2.3					L2-9.3
TCLANM.MAP	FILE	1		1-3.5.1.5					
TCLUNM.MAP	FILE	1		1-3.5.1.5					
TDATE1	COM2 VARIABLE	1		1-3.4.9	1-3.5.2.X				
TDATE2	COM2 VARIABLE	1		1-3.4.9	1-3.5.2.X				
TDATE3	COM2 VARIABLE	1		1-3.4.9	1-3.5.1.3	1-3.5.1.4	1-3.5.2.X		
TDIS	VARIABLE	1		1-3.5.2.6					
THLOPM	SUBROUTINE	1	TASK	1-CONTNTS	1-3.5.2.5				L2-12.8

NAME	DESCRIPTION	LINE COMMENT		OCCURRENCES					LOCATION OF LISTING
THMLOP	SHARE SUBROUTINE	1		1-CONTNTS	3-21.				L3-21.
TI	VARIABLE	1		3-17.					
TIME	F4PLIBSUBROUTINE	1		1-3.5.1.4	1-3.5.2.5	1-3.5.2.6	1-3.5.2.8	1-3.5.2.9	
TOBDUN	VARIABLE	1		1-3.5.1.1					
TP	VARIABLE	1		1-3.5.2.3					
TP1XL	VARIABLE	1		1-3.5.2.9					
TPCT	VARIABLE	1		1-3.5.2.9					
TRAJPL	PRIVT SUBROUTINE	1		1-3.5.2.5					L2-13.8
TSTAT.DAT	FILE	1		1-3.5.1.5					
TTX2	VARIABLE	1		1-3.5.2.3					
TTXI	VARIABLE	1		1-3.5.2.3					
TTY1	VARIABLE	1		1-3.5.2.3					
TTY2	VARIABLE	1		1-3.5.2.3					
TUN	VARIABLE	1		1-3.5.2.3					
TUNIT	VARIABLE	1		1-3.5.2.3					
TWRITE	SHARE SUBROUTINE	1		1-3.5.2.7	2-15.6	3-22.			L2-15.6
		2							L3-22.
TX	VARIABLE	1		1-3.5.2.3	1-3.5.2.7				
TX1	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.3	1-3.5.2.5	1-3.5.2.7		
TX2	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.3	1-3.5.2.5	1-3.5.2.7		
TY	VARIABLE	1		1-3.5.2.3	1-3.5.2.7				
TY1	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.3	1-3.5.2.5	1-3.5.2.7		
TY2	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.3	1-3.5.2.5	1-3.5.2.7		
TYPE	COM5 VARIABLE	1		1-3.4.4	1-3.4.9	1-3.5.2.5	1-3.5.2.6	1-3.5.2.9	
		2		1-3.5.2.X					
TYPE	COM5 VARIABLE	1		1-3.5.2.6	1-3.5.2.9	1-3.5.2.X			
TYPEI	VARIABLE	1		1-3.5.1.3					
TYPI	VARIABLE	1		1-3.5.1.3					
UFLAG1	COM3 VARIABLE	1		1-3.4.9	1-3.5.2.X				
UFLAG2	COM3 VARIABLE	1		1-3.4.9	1-3.5.2.4	1-3.5.2.X			
UFLAG3	COM3 VARIABLE	1		1-3.4.4	1-3.4.9	1-3.5.2.X			
UFLAG4	COM4 VARIABLE	1		1-3.4.9	1-3.5.2.X				
UNBIAS	PRIVT SUBROUTINE	1		1-3.5.2.9					L2-18.3
UNCDIS	PRIVT SUBROUTINE	1		1-3.5.2.7	2-15.2				L2-15.2
UNID	VARIABLE	1		1-3.5.2.8					
UNIT	VARIABLE	1		3-5.					
UNLDOT	PRIVT SUBROUTINE	1		1-3.5.2.X					L2-3
UNPAK	PRIVT FUNCTION	1		1-3.5.1.3					L2-3.1
UNPAK	VARIABLE	1		1-3.5.1.3					
UPCT	VARIABLE	1		1-3.5.2.9					
UPDATE	PRIVT SUBROUTINE	1		1-3.5.1.3	1-3.5.2.6				L2-3.6
UPDOT	PRIVT SUBROUTINE	1		1-3.5.1.2	2-2.15				L2-2.15
U	VARIABLE	1		1-3.5.2.3	3-9.				
VALCK	PRIVT SUBROUTINE	1							L2-12.6
VAR	COM2 VARIABLE	1		1-3.4.9					
VAR0	COM2 VARIABLE	1		1-3.4.9					
VDALTR	SHARE SUBROUTINE	1		1-3.5.2.5	3-23.				L3-23.
VERTEX	VARIABLE	1		1-3.5.2.8	3-11.1				
WAIT	IMALIBSUBROUTINE	1		1-3.5.2.5					
WAITFR	F4PLIBSUBROUTINE	1		1-3.5.1.4	1-3.5.2.1				
WC	VARIABLE	1		1-3.5.2.3					
WINDER	SHARE SUBROUTINE	1		1-CONTNTS	3-24.				L3-24.

NAME	DESCRIPTION	LINE	COMMENT	OCCURRENCES					LOCATION OF LISTING
WINDRM	PROGRAM	1							L2-12.7
WINDRM	SUBROUTINE	1	TASK	1-CONTNTS	1-3.5.2.5				L2-12.
WORK	VARIABLE	1		1-3.5.1.3					
WRDIR PRIVT	SUBROUTINE	1		1-3.5.1.2	2-2.14				L2-2.14
WRDODU PRIVT	SUBROUTINE	1		1-3.5.1.2	2-2.12				L2-2.12
WRDOT PRIVT	SUBROUTINE	1		1-3.5.1.2	2-2.13				L2-2.13
X	VARIABLE	1		1-3.5.2.3	1-3.5.2.8				
XPCT	VARIABLE	1		1-3.5.2.9					
XXXXDOOTS.DAT	FILE	1		1-3.5.2.X					
XXXXDOTS.DAT	FILE	1		1-3.3	1-3.3.3	1-3.4.4	1-3.5.1.3	1-3.5.2.2	
XXXXFIELD.DAT	FILE	1		1-3.3	1-3.3.4	1-3.5.1.2	1-3.5.2.2	1-3.5.2.X	
XXXXPCLAS.MAP	FILE	1		1-3.3	1-3.3.5	1-3.5.2.2	1-3.5.2.X		
XXXXPCLUS.MAP	FILE	1		1-3.3	1-3.3.5	1-3.4.3	1-3.5.2.2	1-3.5.2.X	
XXXXPSTAT.DAT	FILE	1		1-3.3	1-3.3.5	1-3.4.5	1-3.5.2.2	1-3.5.2.X	
XXXXTCLAS.MAP	FILE	1		1-3.3	1-3.3.5	1-3.5.2.2	1-3.5.2.X		
XXXXTCLUS.MAP	FILE	1		1-3.3	1-3.3.5	1-3.4.3	1-3.5.2.2	1-3.5.2.X	
XXXXTSTAT.DAT	FILE	1		1-3.3	1-3.3.5	1-3.4.5	1-3.5.1.4	1-3.5.2.2	
		2		1-3.5.2.X					
XXXXYYDDD.DAT	FILE	1		1-3.3					
XZ	VARIABLE	1		1-3.5.2.3					
Y	VARIABLE	1		1-3.5.2.3	1-3.5.2.5				
YLINE	VARIABLE	1		3-11.2					
YR	VARIABLE	1		1-3.5.1.2	1-3.5.1.3	1-3.5.1.5	1-3.5.2.X		
YZ	VARIABLE	1		1-3.5.2.3					
ZNAME PRIVT	SUBROUTINE	1							L2-5.12
ZD	VARIABLE	1		1-3.5.2.7					
ZDOOM PRIVT	SUBROUTINE	1		1-3.5.2.3	1-3.5.2.7	1-3.5.2.8	2-15.9	3-25.	L2-15.9
		2							+L3-25.